Solving Nonlinear Resistive Networks Using Piecewise-Linear Analysis and Simplicial Subdivision

MING-JEH CHIEN, MEMBER, IEEE, AND ERNEST S. KUH, FELLOW, IEEE

Abstract—**In recent years numerous results of piecewise-linear analysis of nonlinear resistive networks have been derived. The applicability of the method relies on the fact that every nonlinear device is modeled by a piecewise-linear continuous function. In order to extend the applicability of piecewise-linear analysis to treat more general nonlinear networks, three steps need to be carried out:**

i) the subdivision of the domain of the multi-dimensional nonlinear network function;
ii) the interpolation of a piecewise-linear continuous function on the subdivided domain; and
iii) the application of piecewise-linear analysis.

It turns out that the above three steps can be accomplished effectively by using simplicial subdivision. In addition, the difficulties encountered in the conventional piecewise-linear analysis are simplified. The memory space needed for the analysis is also greatly reduced. The complete analysis has been implemented in a program on CDC 6400.

I. Introduction

In 1965, Katzenelson developed an efficient method for solving nonlinear resistive networks which contain uncoupled resistors represented by monotonically increasing piecewise-linear continuous functions [1]. During the past decade, the work has been greatly extended and generalized [2]–[8]. However there exists a fundamental limitation of the approach, that is, nonlinear resistors must be first approximated by continuous piecewise-linear functions.

The purpose of this paper is to develop a method which deals directly with the multidimensional nonlinear network function, thus extending the applicability of the piecewise-linear analysis to solve more general network problems. This method consists of two preliminary steps:

i) the subdivision of the domain of the nonlinear network function; and

ii) the interpolation of a piecewise-linear continuous function on the subdivided domain.

Since the multidimensional piecewise linearization is carried out with respect to the given nonlinear network function, the modeling process of nonlinear resistors is avoided.

In 1956, Stern first proposed the idea of using vertices of simplices for interpolation in the analysis of nonlinear resistive networks [9]. Iri extended Stern's work and furthermore, considered the problem of error estimation [10]. Ohtsuki and Yoshida based on the method above, employed multidimensional interpolation of transistor characteristics in applying Katzenelson's algorithm of piecewise-linear analysis [3]. Our present paper takes into account the latest advances in piecewise-linear analysis [4]–[8], and demonstrates the advantages of multidimensional interpolation by using simplices.

In Section II, we will present a systematic procedure of the simplicial subdivision of an n-dimensional rectangle. The procedure along with pertinent properties of the subdivision is described in detail. In Section III, the piecewise-linear interpolation of nonlinear function based on the simplicial subdivision is discussed. The method of piecewise-linear analysis of nonlinear networks is reviewed and is specially formulated for the purpose of simplifying computation. One important property is that the boundaries of a simplex are simply defined in terms of vertices of the simplex. Consequently, the most tedious part of the conventional piecewise-linear analysis, namely, "the boundary crossing" is easily done by replacing an "old" vertex by a new one. This procedure is treated in Section IV under the title, The Replacement Rule. Thus there is no need of storing information of the specification of regions and boundaries other than verticies of simplices. Therefore the amount of memory space needed is greatly reduced. In Section V, the algorithm is presented together with an example. The method of triangular factorization of modified matrices plays a major role in the computation [6], [11]. The implementation algorithm is programmed on the CDC 6400. Some comments on the program are given in Section VI. Also included is a refined search subprogram to increase accuracy.

Manuscript received August 24, 1976; revised February 4, 1977. This work was supported in part by NSF Grant ENG74 06651-A01 and in part by the Joint Services Electronics Program under Contract F44620-71-C-0087.
M. J. Chien was with the Department of Electrical Engineering and Computer Sciences and the Electronics Research Laboratory, University of California, Berkeley, CA 94720. He is now with the Department of Electrical Engineering, Wayne State University, Detroit, MI.
E. S. Kuh is with the Department of Electrical Engineering and Computer Sciences and the Electronics Research Laboratory, University of California, Berkeley, CA 94720.

II. Simplicial Subdivision

Let $x_0, x_1, \cdots,$ and x_n be $(n+1)$ points in the n-dimensional space. A simplex $S(x_0, \cdots, x_n)$ is defined by

$$S(x_0, \cdots, x_n) = \left\{ x \mid x = \sum_{i=0}^{n} \mu_i x_i, \quad 1 \geqq \mu_i \geqq 0, \quad i = 0,1,2,\cdots,n \text{ and } \sum_{i=0}^{n} \mu_i = 1 \right\}. \quad (1)$$

In other words, $S(x_0, \cdots, x_n)$ is the convex combination of $x_0, x_1, \cdots,$ and x_n which are called the vertices of the simplex $S(x_0, \cdots, x_n)$ [10]–[12]. A simplex $S(x_0, \cdots, x_n)$ is said to be proper if and only if it cannot be contained in an $(n-1)$-dimensional hyperplane $\{x \mid n^T x = \text{constant}\}$. This has been proved to be equivalent to the fact that the $(n+1)\times(n+1)$ matrix $\begin{Bmatrix} x_0 & \cdots & x_n \\ 1 & \cdots & 1 \end{Bmatrix}$ is nonsingular [12]. In this paper, this condition is assumed to hold. A simplex is considered "proper" unless explicitly stated. In a 2-dimensional space $(n=2)$ the above condition simply asserts that $x_0, x_1,$ and x_2 are not on a straight line, as shown in Fig. 1.

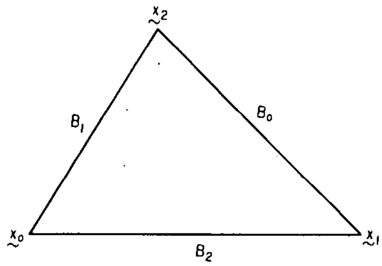


Fig. 1. 2-dimensional proper simplex $S(x_0, x_1, x_2)$ and its boundaries.

Corresponding to the $(n+1)$ vertices, there are $(n+1)$ boundaries. The boundary B_k corresponding to the vertex x_k is defined as

$$B_k = \{x \mid x \in S(x_0, \cdots, x_n) \text{ with } \mu_k = 0\}. \quad (2)$$

It is easy to see that B_k contains all the vertices except x_k. This one-to-one correspondence between vertices and boundaries is shown in Fig. 1 for the case $n=2$. The intersection of more than one boundary is called a corner. Thus a vertex is a corner which is the intersection of n boundaries.

In this paper, we assume that the solution of the nonlinear resistive network is bounded. The determination of such a bounded set, in which the approximate solution lies, has been considered by many authors [20]–[23]. Let the bounded set be contained in an n-dimensional rectangle $RL = \{x \mid a \leqq x \leqq b\}$, where $a < b$.[1] The procedure of subdividing RL consists of two steps, namely, the tessellation of RL into small rectangles and the subdivision of each small rectangle into simplices. For convenience in derviation and computation, each small rectangle in RL is to be transformed to a unit n-cube before subdivision.

The first step is accomplished by the construction of a homeomorphism between RL and $C_p = \{z \mid 0 \leqq z \leqq p, p > 0$ and every component of p is an integer$\}$. The rectangle C_p contains $\Pi_{i=1}^{n} p_i$ n-cubes in the z-space. Let the transformation be defined by

$$z = T(x) = \begin{bmatrix} p_1/(b_1 - a_1) & & 0 \\ & \ddots & \\ 0 & & p_n/(b_n - a_n) \end{bmatrix} (x - a) \quad (3)$$

Accordingly, the rectangle RL is also divided into $\Pi_{i=1}^{n} p_i$ rectangles. Each rectangle is mapped onto one and only one n-dimensional cube in C_p by $T(\cdot)$. This is illustrated in the following example.

Example 1

Let

$$RL = \left\{ x \,\middle|\, \begin{bmatrix} 5.5 \\ -1 \end{bmatrix} \leqq x \leqq \begin{bmatrix} 8.0 \\ 2 \end{bmatrix} \right\}$$

and

$$C_p = \left\{ z \,\middle|\, 0 \leqq z \leqq \begin{bmatrix} 4 \\ 3 \end{bmatrix} \right\}.$$

For any $x \in RL$, there is one and only one $z \in C_p$ defined by

$$z = \begin{bmatrix} 4/(8-5.5) & 0 \\ 0 & 3/(2-(-1)) \end{bmatrix} \left(x - \begin{bmatrix} 5.5 \\ -1 \end{bmatrix} \right).$$

The rectangle RL is divided into 12 small rectangles, as shown in Fig. 2.

Next, we wish to subdivide each small rectangle into simplices. In the z-space, the set of vertices of the cubes contained in C_p is defined by

$$V_p = \{ I \mid 0 \leqq I \leqq p \cdot \text{Every component of } I \text{ is 0 or an integer}, p > 0 \}. \quad (4)$$

From (4), the set of vertices of the rectangles contained in RL in the x-space is $T^{-1}(V_p)$, i.e.,

$$V = T^{-1}(V_p) = \{ x \mid T(x) \in V_p \}. \quad (5)$$

The subdivision of cubes of C_p into "nonoverlapping"

[1] In this paper, we use the following notations:

$x \leqq y$ means $x_i \leqq y_i, \quad i = 1,2,\cdots,n$

$x < y$ means $x \leqq y$ and $x \neq y$

$x < y$ means $x_i < y_i, \quad i = 1,2,\cdots,n$ and

$x \leqq a$ means $x_i \leqq a, \quad i = 1,2,\cdots,n.$

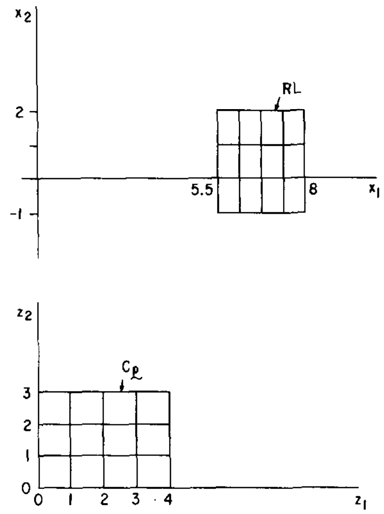


Fig. 2. Tessellation of RL into small rectangles and corresponding tessellation of C_p into unit cubes.

simplices is done by properly arranging the vertices of V_p in a fixed order as shown by the following lemma due to Kuhn [13], [14].

Lemma 1

Every $z \in C_p$ has a unique representation

$$z=\mu_0 I_0+\cdots+\mu_m I_m,$$

where

$$\mu_j>0,\ I_j \in V_p, \qquad \text{for } j=0,1,\cdots,m\left(\leqq n\right),\ \sum_{j=0}^{m}\mu_j=1$$

and

$$I_0 \leqslant I_1 \leqslant \cdots \leqslant I_m \leqq I_0+1.^2$$

Note that the point z is in the cube $C(I_0)$ in C_p, where

$$C(I_0)=\left\{z \mid I_0 \leqq z \leqq I_0+1\right\}.$$

In the case that $m=n$, z is an interior point of the simplex $S(I_0,\cdots,I_n) \subset C(I_0)$. Otherwise, z lies on the boundary of a simplex. The computation of μ_j's and I_j's is illustrated by the following example.

Example 2

Let $n=6$. Let

$$z=\begin{bmatrix}1.3 & 0.6 & 2.9 & 0.4 & 1.5 & 0.8\end{bmatrix}^T.$$

We first decompose z into two parts:

$$z=\begin{bmatrix}0.3 & 0.6 & 0.9 & 0.4 & 0.5 & 0.8\end{bmatrix}^T+I_0$$

[2] I_0+1 denotes a vector which is formed by adding unity to each component of I_0.

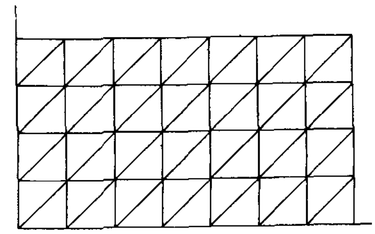


Fig. 3. Canonical decomposition of RL (2-dimensional) into simplices.

where

$$I_0=\begin{bmatrix}1 & 0 & 2 & 0 & 1 & 0\end{bmatrix}^T.$$

The vector $[0.3\ 0.6\ 0.9\ 0.4\ 0.5\ 0.8]^T$ can be represented by

$$0.3\begin{bmatrix}1\\1\\1\\1\\1\\1\end{bmatrix}+0.1\begin{bmatrix}0\\1\\1\\1\\1\\1\end{bmatrix}+0.1\begin{bmatrix}0\\1\\1\\0\\1\\1\end{bmatrix}+0.1\begin{bmatrix}0\\1\\1\\0\\0\\1\end{bmatrix}+0.2\begin{bmatrix}0\\0\\1\\0\\0\\1\end{bmatrix}+0.1\begin{bmatrix}0\\0\\1\\0\\0\\0\end{bmatrix}+0.1\begin{bmatrix}0\\0\\0\\0\\0\\0\end{bmatrix}.$$

Therefore

$$z=\begin{bmatrix}1.3\\0.6\\2.9\\0.4\\1.5\\0.8\end{bmatrix}=0.3\begin{bmatrix}2\\1\\3\\1\\2\\1\end{bmatrix}+0.1\begin{bmatrix}1\\1\\3\\1\\2\\1\end{bmatrix}+0.1\begin{bmatrix}1\\1\\3\\0\\2\\1\end{bmatrix}$$

$$+0.1\begin{bmatrix}1\\1\\3\\0\\1\\1\end{bmatrix}+0.2\begin{bmatrix}1\\0\\3\\0\\1\\1\end{bmatrix}+0.1\begin{bmatrix}1\\0\\3\\0\\1\\0\end{bmatrix}+0.1\begin{bmatrix}1\\0\\2\\0\\1\\0\end{bmatrix}$$

$$=\mu_6 I_6+\mu_5 I_5+\mu_4 I_4+\mu_3 I_3+\mu_2 I_2+\mu_1 I_1+\mu_0 I_0.$$

The conditions of Lemma 1 are fulfilled, namely,

i) $\mu_0>0,\ \mu_1>0,\cdots,\mu_6>0$
ii) $\sum_{j=0}^{6}\mu_j=1$ and
iii) $I_0 \leqslant I_1 \leqslant I_2 \leqslant I_3 \leqslant I_4 \leqslant I_5 \leqslant I_6=I_0+1.$

The result in Lemma 1 provides a canonical decomposition of the rectangle C_p into simplices. The decomposition for $n=2$ is shown in Fig. 3. Each 2-cube (square) is divided into two 2-dimensional simplices. Each simplex

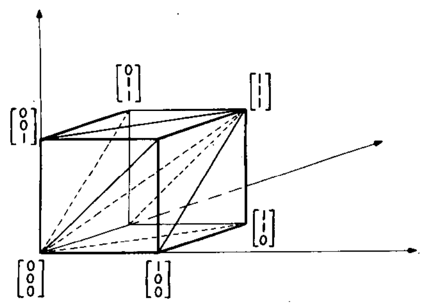


Fig. 4. Canonical decomposition of 3-dimensional cube into simplices.

contains three vertices. The decomposition of a unit 3-cube is shown in Fig. 4. A unit 3-cube is divided into six 3-dimensional simplices. Each simplex contains four vertices. In general, the n-dimensional simplices generated by the above method, called the simplicial subdivision, have the following properties.

i) The union of these simplices is C_p.

ii) They are "nonoverlapping;" more specifically, if the intersection of any two simplicies is nonempty, then it is either a boundary or a corner.

iii) Every n-dimensional simplex defined by $(n+1)$ vertices contains I_0 and I_0+1 which define the cube $C(I_0)$ containing the simplex.

Consequently, RL in the x-space is also divided into simplices which have the above properties. It should be mentioned that the homeomorphism $T(\cdot)$ preserves the ordering of vertices, since the matrix defined in (3) is a positive-definite diagonal matrix. In other words, $I_i \leqslant I_j$ if and only if $x_i \leqslant x_j$, where I_i, $I_j \in V_p$, $I_i = T(x_i)$ and $I_j = T(x_j)$. Thus, from Lemma 1, we obtain the following.

Lemma 2

Every $x \in RL$ has a unique representation

$$x = \mu_0 x_0 + \cdots + \mu_m x_m$$

where

$$\mu_j > 0,\ x_j \in V, \qquad \text{for } j = 0, 1, \cdots, m\left(\leqq n\right),\ \sum_{j=0}^{m} \mu_j = 1$$

and

$$T(x_0) \leqslant T(x_1) \leqslant \cdots \leqslant T(x_m) \leqq T(x_0) + 1.$$

III. Piecewise-Linear Analysis

Having presented the simplicial subdivision of an n-dimensional rectangle RL, we are ready to take a look at piecewise-linear interpolation. Let the nonlinear equation be $g(x) = y$, where $g(\cdot)$ maps from R^n into itself. Let $S(x_0, \cdots, x_n)$ be any simplex generated by the simplicial subdivision. An affine function approximating the given $g(\cdot)$ on $S(x_0, \cdots, x_n)$ can be defined by

$$f(x) = [\, g(x_0), \cdots, g(x_n)\,]\, \mu \tag{6}$$

for $x \in S(x_0, \cdots, x_n)$ and $\mu = [\mu_0, \mu_1, \cdots, \mu_n]^T$ as in (1). Extending this interpolating procedure to all simplices, we have a piecewise-linear continuous function $f(\cdot)$ approximating $g(\cdot)$, which is defined on the rectangle RL. The continuity of $f(\cdot)$ is a direct consequence of the fact that the simplices do not overlap.

Suppose the function $g(\cdot)$ is continuous in RL, then there exists $\epsilon > 0$ such that $\| g(x) - g(x') \| \leqq \epsilon$ for all x, $x' \in S(x_0, \cdots, x_n) \subset RL$. This together with (1) leads to the following lemma.

Lemma 3

If $g(\cdot)$ is continuous in RL which is the union of nonoverlapping simplices, then

$$\| f(x) - g(x) \| \leqq \epsilon, \qquad \text{for all } x \in RL$$

where

$$\epsilon = \max_{S \in RL} \max \{ \| g(x) - g(x') \|; \; x \text{ and } x' \text{ are in simplex } S \}$$

and $f(\cdot)$ in each simplex is the interpolating function defined in (6).

Proof: For every $x \in RL$

$$x \in S(x_0, x_1, \cdots, x_n) \subset RL$$

$$f(x) = \sum_{i=0}^{n} \mu_i g(x_i)$$

where

$$\mu_i \geqq 0, \qquad i = 0, 1, \cdots, n$$

and

$$\sum_{i=0}^{n} \mu_i = 1.$$

Thus

$$\| f(x) - g(x) \| = \left\| \sum_{i=0}^{n} \mu_i \big(g(x_i) - g(x) \big) \right\|$$

$$\leqq \sum_{i=0}^{n} \mu_i \| g(x_i) - g(x) \| \leqq \epsilon$$

since x_i and x are in the same simplex $S(x_0, x_1, \cdots, x_n)$.

In this paper we shall adopt the following representation:

$$\begin{bmatrix} f(x) \\ 1 \end{bmatrix} = \begin{bmatrix} g(x_0) & \cdots & g(x_n) \\ 1 & \cdots & 1 \end{bmatrix} \mu \triangleq G\mu = \begin{bmatrix} y \\ 1 \end{bmatrix} \tag{7}$$

for

$$x \in S(x_0, \cdots, x_n) = \left\{ x \left| \begin{bmatrix} x \\ 1 \end{bmatrix} = \begin{bmatrix} x_0 & \cdots & x_n \\ 1 & \cdots & 1 \end{bmatrix} \mu \right. \right\}.$$

This is similar to the Wolfe secant formulation [12]. If $S(x_0, \cdots, x_n)$ is proper, (7) can also be written as

$$\begin{bmatrix} f(x) \\ 1 \end{bmatrix} = \begin{bmatrix} g(x_0) & \cdots & g(x_n) \\ 1 & \cdots & 1 \end{bmatrix} \begin{bmatrix} x_0 & \cdots & x_n \\ 1 & \cdots & 1 \end{bmatrix}^{-1} \cdot \begin{bmatrix} x \\ 1 \end{bmatrix} \triangleq J \begin{bmatrix} x \\ 1 \end{bmatrix} = \begin{bmatrix} y \\ 1 \end{bmatrix} \tag{8}$$

where J is the Jacobian matrix in the usual formulation of

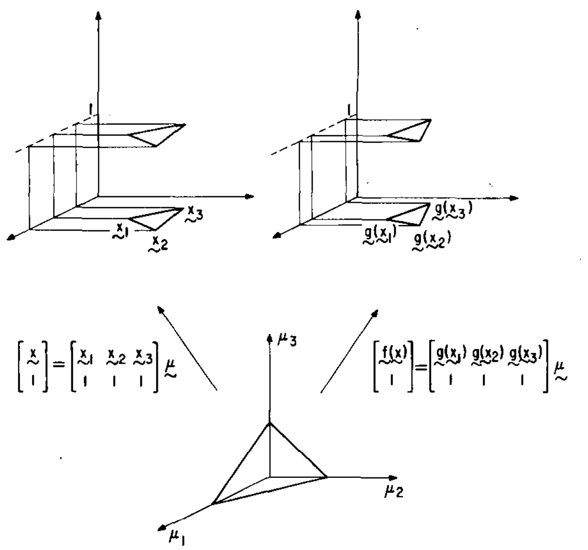


Fig. 5. Relations among various spaces as given in (7) and (8).

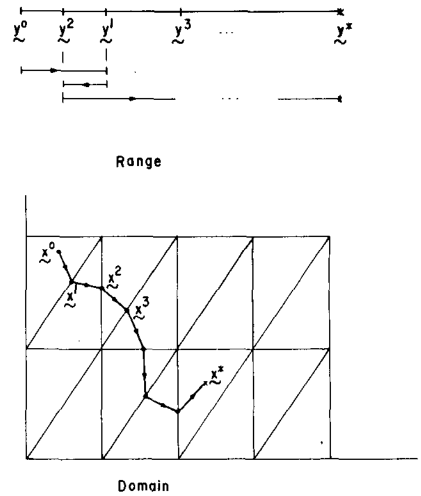


Fig. 6. Solution curve in domain, $L(x^0)$, enters new region at each iteration and converges to solution x^* as y^* is reached.

the piecewise-linear analysis. The geometric interpretation of (7) and (8) is shown in Fig. 5 for $n=2$.

The advantages of this particular formulation in (7) are the following.

i) There is no need to compute the Jacobian matrix J.

ii) Since $x \in S(x_0, \cdots, x_n)$ if and only if the vector μ satisfies $1 \geqq \mu \geqq 0$, it is very easy to check whether an approximate solution is found in the simplex. More specifically, an approximate solution of the equation, $g(x) = y$, is found in $S(x_0, \cdots, x_n)$ if and only if the solution of (7) satisfies $\mu \geqq 0$.

We next review briefly the method of piecewise-linear analysis. Consider an arbitrary nonlinear resistive network. Let the vector x represent the chosen network variables and the vector y the inputs to the network. It is well known that a nonlinear resistive network can be described by [5,6]

$$g(x)=y \tag{9}$$

where $g(\cdot)$ is a continuous function from R^n into itself. In the piecewise-linear analysis, the continuous function $g(\cdot)$ is piecewise linear, that is,

$$g(x)=J^{(m)}x+w^{(m)}=y, \qquad m=0,1,\cdots,l \tag{10}$$

where $J^{(m)}$ is a constant Jacobian matrix and $w^{(m)}$ is a constant vector defined in the region R_m. The finite integer l denotes that the total number of regions is finite. The piecewise-linear analysis amounts to finding a continuous piecewise-linear curve from an initial point x^0 to a solution in the domain space such that the image of this curve is a straight line which connects $y^0 = g(x^0)$ and the given input $y = y^*$ [4]–[8].

The continuous piecewise-linear curve, $L(x^0)$, is called the solution curve in the domain. The image of $L(x^0)$, $L(y^0)$, is called the solution curve in the range space. The computation of both solution curves is done by an iterative procedure. More specifically, a series of doublets, (x^i, y^i), $i=0,1,2,\cdots$, is calculated such that

i) $y^i = g(x^i)$

ii) $\{x \mid x = x^i + t(x^{i+1} - x^i)\ t \in [0,1]\}$ is in the region R_i, $i=0,1,2,\cdots$, and

iii) $y^0, y^1, y^2 \cdots$ are on a straight line.

A number of conditions have been derived, which guarantee that the sequence, $y^0, y^1, y^2, \cdots$, converges to y^* in a finite number of steps. Accordingly, the sequence in the domain, $x^0, x^1, x^2, \cdots$, converges to a point x^* which is a solution of (10). This is shown in Fig. 6. The main feature of this approach, which should be emphasized, is that the solution curve $L(x^0)$ enters a new region at each iteration.

The operations which are needed to carry out the analysis are summarized as follows.

i) The determination of the boundary of the present region to be crossed. (When a solution cannot be found in the present region, the solution curve $L(x^0)$ traverses the region and reaches a boundary which is to be crossed.)

ii) The identification of the new region into which the solution curve should enter. (This new region is uniquely defined when the solution curve reaches a single boundary.)

iii) The formulation of the new equation to compute the next segments of the solution curves, $L(x^0)$ and $L(y^0)$.

When the simplicial subdivision is used to divide the domain into simplices and a continuous piecewise-linear function $f(\cdot)$ is interpolated on those simplices, the above three steps are greatly simplified as follows.

i) The determination of the boundary B_k to be crossed is equivalent to the determination of the vertex x_k to be deleted from the present simplex $S(x_0, \cdots, x_n)$. As discussed in the previous section, x_k is opposite to B_k.

ii) The computation of the vertex x_k' which forms, together with the remaining n vertices, the new simplex $S(x_0, \cdots, x_{k-1}, x_k', x_{k+1}, \cdots, x_n)$. The solution curve is forced to enter this new simplex.

iii) The new equation is simply formulated by replacing the $(k+1)$th column of the matrix G in (7) by $\begin{bmatrix} g(x_k') \\ 1 \end{bmatrix}$, that is,

$$\begin{bmatrix} g(x_0) & \cdots & g(x_{k-1}) & g(x_k') & g(x_{k+1}) & \cdots & g(x_n) \\ 1 & \cdots & 1 & 1 & 1 & \cdots & 1 \end{bmatrix}. \tag{11}$$

The procedure to replace x_k by x_k' is called the replacement rule and is presented in the next section.

IV. The Replacement Rule

The derivation of the replacement rule relies on the fact that the generated simplices are "nonoverlapping," i.e., if the intersection of two simplices is not empty, then it is either a boundary of the simplices or a corner. Furthermore, for any given boundary of a simplex, there are two and only two simplices containing the boundary, the only exception being that the boundary is itself a subset of the boundary of the rectangle RL. When a solution is not found in the present simplex $S(x_0,\cdots,x_n)$, the solution curve traverses $S(x_0,\cdots,x_n)$ and reaches either a boundary or a corner. Let the solution curve $L(x^0)$ reach a boundary but not a corner. It is easy to see that the next simplex is uniquely defined because there are only two simplices which contain the boundary. In other words, if the solution curve is forced to enter a new simplex at each iteration, and if the solution curve $L(x^0)$ reaches a boundary but not a corner, then the new simplex at each iteration is completely determined by the structure of the simplices.

Let the solution curve $L(x^0)$, traverse $S(x_0,\cdots,x_n)$ and reach a point x^i on the boundary

$$B_k=\left\{x \middle| \begin{bmatrix} x \\ 1 \end{bmatrix} = \begin{bmatrix} x_0 & \cdots & x_n \\ 1 & \cdots & 1 \end{bmatrix}\mu, \text{ the } k\text{th component of } \mu, \mu_k, \text{ is zero}\right\}.$$

The new region is determined by the boundary B_k and a new vertex x_k' which is computed according to the following theorem.

Theorem 1

Let $T(x_0)\leqslant T(x_1)\leqslant \cdots \leqslant T(x_n)=T(x_0)+1$, where $x_j\in V, j=0,1,\cdots,n$. Then the new vertex x_k' is defined by

$$x_{k+1}+x_{k-1}-x_k, \qquad k=0,1,\cdots,n,$$

where $x_{n+1}\triangleq x_0$ and $x_{-1}\triangleq x_n$.

Before presenting the proof, we illustrate the geometrical meaning of this theorem by the following example.

Example 3

Let

$$RL=\left\{x \middle| \begin{bmatrix} 5.5 \\ -1 \end{bmatrix} \leqq x \leqq \begin{bmatrix} 8 \\ 2 \end{bmatrix}\right\}$$

and

$$C_p=\{z|0\leqq z\leqq 4\}$$

as shown in Fig. 7. Let the present simplex be defined by

$$x_0=\begin{bmatrix} 5.5+2\cdot\left(\frac{8-5.5}{4}\right) \\ -1+2\cdot\left(\frac{2+1}{4}\right) \end{bmatrix}$$

$$x_1=\begin{bmatrix} 5.5+3\cdot\left(\frac{8-5.5}{4}\right) \\ -1+2\cdot\left(\frac{2+1}{4}\right) \end{bmatrix}$$

and

$$x_2=\begin{bmatrix} 5.5+3\cdot\left(\frac{8-5.5}{4}\right) \\ -1+3\cdot\left(\frac{2+1}{4}\right) \end{bmatrix}.$$

Suppose x_1 is deleted, the new vertex x_1' is computed as

$$x_1'=x_2+x_0-x_1=\begin{bmatrix} 5.5+2\cdot\left(\frac{8-5.5}{4}\right) \\ -1+3\cdot\left(\frac{2+1}{4}\right) \end{bmatrix}.$$

Proof of Theorem 1: Since $x^i\in B_k$, it is uniquely represented by $x^i=\sum_{j=0}^{n}\mu_j^i x_j$ where $\mu_k^i=0$ and $1>\mu_j^i>0$ for $j\neq k$. Thus x^i is an interior point of the boundary hyperplane B_k. Let ϵ be an "arbitrarily small" positive number. In Appendix A, it is shown that the vector

$$x(\epsilon)=\sum_{\substack{j=0\\ j\neq(k-1),k\\ (k+1)}}^{n} \mu_j^i x_j+\left(\mu_{k-1}^i+\frac{\epsilon}{2}\right)x_{k-1} - \epsilon x_k+\left(\mu_{k+1}^i+\frac{\epsilon}{2}\right)x_{k+1} \tag{12}$$

defines a point in a new simplex. Equation (12) can be written as

$$x(\epsilon)=\sum_{\substack{j=0\\ j\neq(k-1),k\\ (k+1)}}^{n} \mu_j^i x_j+\left(\mu_{k-1}^i-\frac{\epsilon}{2}\right)x_{k-1} +\epsilon(x_{k+1}+x_{k-1}-x_k)+\left(\mu_{k+1}^i-\frac{\epsilon}{2}\right)x_{k+1}. \tag{13}$$

From the discussion in Section II, it is clear that $x_k'=x_{k+1}+x_{k-1}-x_k$ is also a vertex of RL in the set V. The

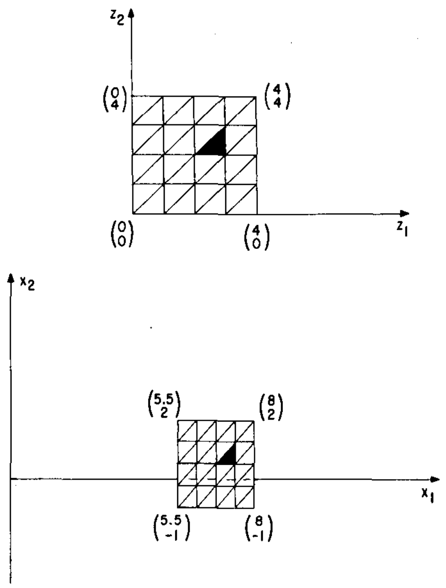


Fig. 7. Illustration of Theorem 1 (Example 3).

conditions of Lemma 2 are fulfilled when ϵ is sufficiently small, for example $\mu_j > 0$ and $\Sigma_{j=0}^{n} \mu_j = 1$; hence; the representation (13) is unique. Note that in (13) $x(\epsilon) \to x^i$ as $\epsilon \to 0$. Thus $x(\epsilon)$ is an interior point of the new simplex when ϵ is sufficiently small. Consequently, the vertex replacing x_k is defined by $x_k' = x_{k+1} + x_{k-1} - x_k$. The same argument holds for $k=0$ or n. The only difference is that the "new" simplex is in a new rectangle. This completes the proof.

V. The Algorithm

With the replacement rule, the new region which the solution curve enters is easily determined. The equation for the new region is

$$\begin{bmatrix} g(x_0) & \cdots & g(x_{k-1}) & g(x_k') & g(x_{k+1}) & \cdots & g(x_n) \\ 1 & \cdots & 1 & 1 & 1 & \cdots & 1 \end{bmatrix} \cdot [\mu] = \begin{bmatrix} y^* \\ 1 \end{bmatrix} \quad (14)$$

where y^* denotes the input. Let the solution of (14) be $\hat{\mu}$. If all the components of $\hat{\mu}$ are larger than or equal to zero, a solution x^* is found in this simplex,

$$x^* = [x_0 \cdots x_{k-1} x_k' x_{k+1} \cdots x_n]\, \hat{\mu}. \quad (15)$$

The segment of the solution curve $L(x^0)$ in this simplex is simply obtained by connecting x^i and x^*, i.e.,

$$x(t) = x^i + t(x^* - x^i), \qquad 1 \geqq t \geqq 0. \quad (16)$$

In the case that $\hat{\mu}$ defines a point outside the simplex, two steps need to be taken to continue the tracing of the solution curve. From the previous discussion,

$$x^i = [x_0, \cdots, x_{k-1}, x_k', x_{k+1}, \cdots, x_n]\, \mu^i$$

is on the boundary

$$B_k = \left\{ x \middle| \begin{bmatrix} x \\ 1 \end{bmatrix} = \begin{bmatrix} x_0, & \cdots, & x_{k-1}, & x_k', & x_{k+1}, & \cdots, & x_n \\ 1, & \cdots, & 1, & 1, & 1, & \cdots, & 1 \end{bmatrix} \mu, \quad \text{the } k\text{th component of } \mu, \mu_k, \text{ is zero} \right\}$$

hence, $\mu_k^i = 0$. Consequently, the segment of $L(x^0)$ entering the new simplex is defined by

$$x(t) = [x_0, \cdots, x_{k-1}, x_k', x_{k+1}, \cdots, x_k]\, \mu(t), \qquad t > 0$$

where

$$\mu(t) = \mu^i + \operatorname{sgn}(\hat{\mu}_k) \cdot t \cdot (\hat{\mu} - \mu^i). \quad (17)$$

The function sgn $(\cdot)$ is defined as

$$\begin{aligned} \operatorname{sgn}(a) &= 1, \qquad \text{if } a \geqslant 0 \\ &= -1, \qquad \text{if } a < 0. \end{aligned}$$

Notice that if x^i is on a single boundary, i.e., $\mu_j^i > 0, j \neq k$, then $\mu(t) > 0$ when $t > 0$ is small enough.

The second step is the computation of $\lambda^i > 0$ such that $\mu(\lambda^i)$ defines a point on the next boundary of the simplex. This is accomplished by checking the components of $\mu(t)$. At least one component of $\mu(t)$ will approach zero when the value of t increases from zero. This fact is derived from the following lemma.

Lemma 4

At least one component of $(\hat{\mu} - \mu^i)$ is positive (negative) if $\hat{\mu} \neq \mu^i$. Thus

$\lambda^i > 0$ is computed by

$$\lambda^i = \min_j \left\{ \frac{-\mu_j^i}{\operatorname{sgn}(\hat{\mu}_k) \cdot (\hat{\mu}_j - \mu_j^i)} > 0 \right\}.$$

The vector $\mu(\lambda^i)$ defines a point on a boundary of the simplex, namely,

$$x^{i+1} = [x_0 \cdots x_{k-1} x_k' x_{k+1} \cdots x_n]\, \mu(\lambda^i).$$

The solution curve $L(x^0)$ in this simplex is then defined by

$$\{ x | x = x^i + t(x^{i+1} - x^i), \qquad 1 \geqq t \geqq 0 \}.$$

The computation of λ^i also determines an index which identifies the vertex to be deleted from the simplex. The procedure discussed above is repeated until a solution is found.

The following algorithm summarizes the above discussion.

Algorithm I

Step 1: Choose x_0 and

$$x_i = x_{i-1} + E_i, \quad i = 1, 2, \cdots, n,$$

where

$E_i = [0, \cdots, 0, e_i, 0, \cdots, 0]^T$ and $e_i > 0$ is the ith component of E_i.

Step 2: Let $\mu^0 = 1/(n+1)[1, \cdots, 1]^T$, i.e.,

$$x^0 = \frac{1}{n+1} \sum_{i=0}^{n} x_i$$

which is the center of the initial simplex. Set $i=0$ and sgn $(\hat{\mu}_k^0)=1$.

Step 3: Compute $\hat{\mu}^i$ according to the equation

$$\begin{bmatrix} g(x_0) & \cdots & g(x_n) \\ 1 & \cdots & 1 \end{bmatrix} \hat{\mu}^i = \begin{bmatrix} y^* \\ 1 \end{bmatrix}.$$

If every component of $\hat{\mu}^i$ is nonnegative, a solution

$x^* = [x_0, \cdots, x_n]\, \hat{\mu}^i$ is found. STOP

Step 4: Otherwise, compute λ^i from

$$\mu(t) = \mu^i + \operatorname{sgn}(\hat{\mu}_k^i) \cdot t \cdot (\hat{\mu}^i - \mu^i)$$

such that

i) $0 \leqq \mu(t) \leqq 1$ for $0 \leqq t \leqq \lambda^i$

ii) there is one and only one index k satisfying $\mu(\lambda^i)_k = 0$

iii) $1 > \mu(\lambda^i)_j > 0$ for $j \neq k$.

Let $\mu^{i+1} = \mu(\lambda^i)$.

Step 5: Replace

$$x_k \text{ by } (x_{k+1} + x_{k-1} - x_k).$$

If this new vertex is outside of the interested range (RL) the algorithm is terminated. Otherwise let $i = i+1$ and go to Step 3.

The convergence of Algorithm I depends completely on the continuous piecewise-linear function $f(\cdot)$ interpolated on the rectangle RL. Let $B(RL)$ denote the boundary of $RL = \{x | a \leqq x \leqq b\}$. Then the algorithm will locate a solution in a finite number of steps under the following conditions:

i) the Jacobian matrix

$$\mathbf{G} = \begin{bmatrix} g(x_0) & \cdots & g(x_n) \\ 1 & \cdots & 1 \end{bmatrix}$$

is nonsingular at each iteration;

ii) the solution curve in the range does not go back to $y^0 = f(x^0)$;

iii) the solution curve in the domain never reaches a corner;

iv) the solution curve does not reach $B(RL)$.

Conditions i)–iii) guarantee that the solution curve in the domain is well determined at each iteration. Condition ii) and iii) further exclude the possibility that the solution curve in the domain becomes cyclic. Condition iv) asserts that the solution curve stays in RL before a solution is found. Two possible situations which violate condition iv) are shown in Figs. 8 and 9. If the image of $B(RL)$ is not the boundary of $f(RL)$, then the solution curve might reach $B(RL)$, and hence, the algorithm is terminated without locating a solution as shown in Fig. 8. The same problem might occur if $L(y^0, y^*)$ is not contained in $f(RL)$ as shown in Fig. 9.

Also, it should be pointed out that in the first iteration, the condition $x_0 \leqslant x_1 \leqslant \cdots \leqslant x_n$ is satisfied, i.e., the vertices obey an ordering property. In step 5, x_k is replaced by $(x_{k+1} + x_{k-1} - x_k)$ without rearranging the vertices. This is true even in the case that $k=0$ or n. The reason is that Theorem 1 is applicable to a "circular" list of vertices. This property is very important from a computational point of view. If a rearrangement of vertices is necessary, the method of matrix modification cannot be applied [6], [11]. In [6], an algorithm was developed based on the fact that the "new" Jacobian matrix is a dyad modification of the "old" Jacobian matrix. In the present case the Jacobian matrix in the "new" simplex $S(x_0, x_1, \cdots, x_{k-1}, x_k', x_{k+1}, \cdots, x_m)$ is indeed a dyad modification of the old one. Therefore, the algorithm developed in [6] can be readily used here.

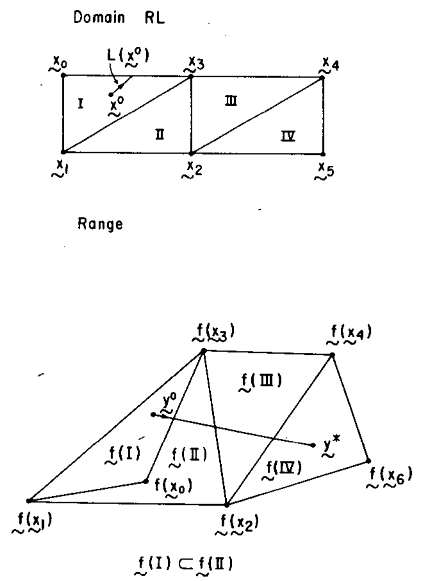


Fig. 8. Solution curve $L(x^0)$ reaches boundary of RL.

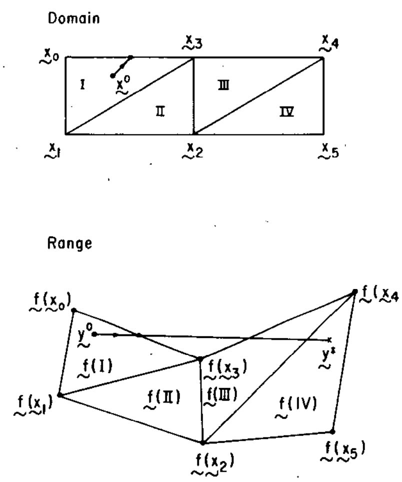


Fig. 9. Line segment $L(y^0, y^*)$ is not contained in $f(RL)$.

Example 4

To illustrate Algorithm I we consider the tunnel diode circuit, as shown in Fig. 10.

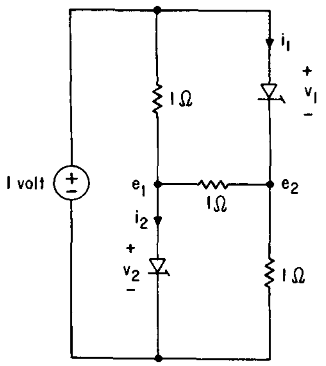


Fig. 10. Tunnel diode circuit in Example 4.

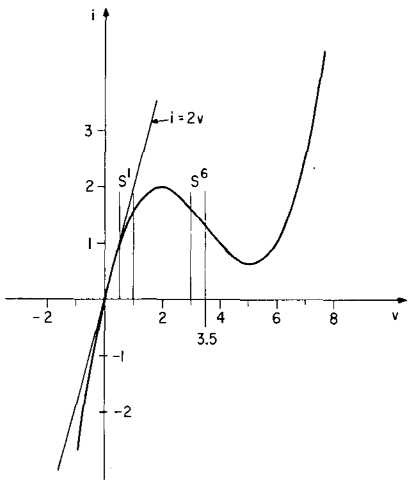


Fig. 11. Model of tunnel diode.

The network equation describing the circuit is

$$2e_1 - e_2 + i_2(e_1) - 1 = y_1 = 0$$
$$-e_1 + 2e_2 - i_1(1 - e_2) = y_2 = 0$$

where e_1 and e_2 are node voltages as shown in the figure and are the components of the vector $\boldsymbol{x}$. The tunnel diodes are represented by

$$i(v) = \frac{29}{12}v - \frac{7}{8}v^2 + \frac{1}{12}v^3$$

as shown in Fig. 11. Let

$$\boldsymbol{x}_0 = \begin{pmatrix} -0.5 \\ 0 \end{pmatrix}$$

$$\boldsymbol{x}_1 = \boldsymbol{x}_0 + \begin{pmatrix} 0.5 \\ 0 \end{pmatrix} = \begin{pmatrix} 0 \\ 0 \end{pmatrix}$$

$$\boldsymbol{x}_2 = \boldsymbol{x}_1 + \begin{pmatrix} 0 \\ 0.5 \end{pmatrix} = \begin{pmatrix} 0 \\ 0.5 \end{pmatrix}.$$

The equation to be solved at the first iteration is

$$\begin{bmatrix} \boldsymbol{g}(\boldsymbol{x}_0) & \boldsymbol{g}(\boldsymbol{x}_1) & \boldsymbol{g}(\boldsymbol{x}_2) \\ 1 & 1 & 1 \end{bmatrix} \hat{\boldsymbol{\mu}} = \begin{bmatrix} -3.4375 & -1 & -1.5 \\ -1.125 & -1.625 & 0 \\ 1 & 1 & 1 \end{bmatrix} \hat{\boldsymbol{\mu}} = \begin{bmatrix} 0 \\ 0 \\ 1 \end{bmatrix}.$$

The solution is

$$\hat{\boldsymbol{\mu}}^0 = \begin{bmatrix} -0.657 \\ 0.455 \\ 1.202 \end{bmatrix}.$$

Since $\boldsymbol{\mu}_0^0$ is the only component which satisfies

$$\frac{-\mu_0^0}{\operatorname{sgn}(\hat{\mu}_0^0)\cdot(\hat{\mu}_0^0 - \mu_0^0)} > 0, \qquad (k=0)$$

the vertex $\begin{bmatrix} -0.5 \\ 0 \end{bmatrix}$ is replaced by

$$\begin{bmatrix} 0 \\ 0 \end{bmatrix} + \begin{bmatrix} 0 \\ 0.5 \end{bmatrix} - \begin{bmatrix} -0.5 \\ 0 \end{bmatrix} = \begin{bmatrix} 0.5 \\ 0.5 \end{bmatrix}.$$

The new simplex is defined by

$$\begin{bmatrix} 0.5 \\ 0.5 \end{bmatrix}, \begin{bmatrix} 0 \\ 0 \end{bmatrix}, \begin{bmatrix} 0 \\ 0.5 \end{bmatrix}.$$

Note that the determination of the new simplex does not require the computation of intersection of the boundary hyperplane and the solution curve. The new equation is

$$\begin{bmatrix} 0.5 & -1 & -1.5 \\ -0.5 & -1.625 & 0 \\ 1 & 1 & 1 \end{bmatrix} \hat{\boldsymbol{\mu}} = \begin{bmatrix} 0 \\ 0 \\ 1 \end{bmatrix}.$$

The solution is

$$\hat{\boldsymbol{\mu}}^1 = \begin{bmatrix} 0.8125 \\ -0.25 \\ 0.4375 \end{bmatrix}.$$

It is easy to see that the vertex $\begin{bmatrix} 0 \\ 0 \end{bmatrix}$ has to be replaced by $\begin{bmatrix} 0.5 \\ 1 \end{bmatrix}$. The new simplex is then defined by

$$\begin{bmatrix} 0.5 \\ 0.5 \end{bmatrix}, \begin{bmatrix} 0.5 \\ 1 \end{bmatrix}, \begin{bmatrix} 0 \\ 0.5 \end{bmatrix}$$

and the new equation is

$$\begin{bmatrix} 0.5 & 0 & -1.5 \\ -0.5 & 1.5 & 0 \\ 1 & 1 & 1 \end{bmatrix} \hat{\boldsymbol{\mu}} = \begin{bmatrix} 0 \\ 0 \\ 1 \end{bmatrix}.$$

The solution is $\hat{\boldsymbol{\mu}}^2 = \begin{bmatrix} 0.6 \\ 0.2 \\ 0.2 \end{bmatrix} > 0$. The approximate solution is found to be

$$\begin{bmatrix} e_1 \\ e_2 \end{bmatrix} = \begin{bmatrix} 0.5 & 0.5 & 0 \\ 0.5 & 1 & 0.5 \end{bmatrix} \begin{bmatrix} 0.6 \\ 0.2 \\ 0.2 \end{bmatrix} = \begin{bmatrix} 0.4 \\ 0.6 \end{bmatrix}.$$

Substituting $\begin{bmatrix} 0.4 \\ 0.6 \end{bmatrix}$ into the network equation, we find

$$y = \begin{bmatrix} y_1 \\ y_2 \end{bmatrix} = \begin{bmatrix} 0.03 \\ -0.03 \end{bmatrix}$$

whereas the actual input is $y^* = \begin{bmatrix} 0 \\ 0 \end{bmatrix}$.

VI. The Program

Algorithm I has been implemented. The complete program is written in Fortran IV language on CDC 6400 [24]. To start with, an n-dimensional rectangle $RL=\{\boldsymbol{x}|\boldsymbol{a} \leqq \boldsymbol{x} \leqq \boldsymbol{b}\}$ is picked, where $\boldsymbol{a}$ and $\boldsymbol{b}$ are referred as the lower bound and the upper bound, respectively. Uniform subdivision of RL is made. The accuracy of the solution of course depends on the grid size, that is, the size of the subdivided n-cube. In this program, the number of subdivisions in each axis is 9, 10 or 11, and we refer to the number as the grid number. For any given function, the inputs to the program are therefore the lower bound, the upper bound, the grid number and the initial grid. These are illustrated in Fig. 12 for the 2-dimensional case.

The program then searches through the domain. The accuracy of the solution is checked by substituting each solution into the equation $\boldsymbol{g}(\boldsymbol{x})=\boldsymbol{y}^*$. If the error of the calculated $\boldsymbol{g}(\boldsymbol{x})$ is satisfactory, for example, within 1 percent of $\boldsymbol{y}^*$, then the program terminates. Otherwise, a refined search is next undertaken with a new domain defined by the grid in which the solution lies.

Example 5

Consider the nonlinear network shown in Fig. 13. The nodal equations are given below:

$$g_1(\boldsymbol{e})=0.1e_1^3-e_1^2+3e_1-e_2-e_3=y_1^*$$
$$g_2(\boldsymbol{e})=0.1e_2^3-e_2^2+3e_2-e_1-e_4=y_2^*$$
$$g_3(\boldsymbol{e})=e_3^3+2e_3^2-e_2-e_4=y_3^*$$
$$g_4(\boldsymbol{e})=e_4^3+2e_4^2-e_1-e_3=y_4^*.$$

Let the input current be $(2,0,2,0)^T$. The solution obtained after two refined searches is $(8.089,7.787,1.974,1.852)^T$. The calculated input is $(2,-0.00046,1.998,-0.0024)^T$. The detailed calculation is given in the print-out in Appendix B.

VIII. Conclusion

In this paper, the method of simplicial subdivision and its application to piecewise-linear analysis are presented. The method of subdivision provides a systematic way to tessellate an n-dimensional rectangle into simplices. Interpolation is then applied to each simplex. When applied to piecewise-linear analysis, the transition from one region to another is extremely simple and is determined by the structure of the simplices. In fact, the procedure is equivalent to the deletion of a vertex from the present simplex and the identification of a vertex to form the new simplex. Consequently, the most difficult part of the piecewise-linear analysis, that is, the boundary crossing has been overcome. Most of the theorems and computation techniques in piecewise-linear analysis can be applied to this new formulation.

Further study is needed in order to make the method more effective. Investigation on the determination of the bounded set in order to select an n-dimensional rectangle RL for a given equation is important.

Appendix A

The purpose of this appendix is to demonstrate that (12) defines a point in a new simplex if ϵ is an arbitrarily small positive number. Let

$$A=\left\{\boldsymbol{x}|\boldsymbol{x}=\sum_{j=0}^{n}\mu_j\boldsymbol{x}_j,\ \sum_{j=0}^{n}\mu_j=1\right\}$$

where $\boldsymbol{x}_0,\boldsymbol{x}_1,\cdots,\boldsymbol{x}_n$ form a proper simplex in the n-dimensional space. Then

$$H_k=\{\boldsymbol{x}|\boldsymbol{x}\in A \text{ and } \mu_k=0\}$$

is an $(n-1)$-dimensional hyperplane. Let $\boldsymbol{n}_k$ be the unit normal vector of the hyperplane H_k, its direction can be chosen such that $d_k=\boldsymbol{n}_k^T\boldsymbol{x}$, for any $\boldsymbol{x}\in H_k$, be positive. Thus d_k represents the distance from the origin to the hyperplane H_k.

Proposition 1: For any $\boldsymbol{x}\in A$

$$\mu_k(\boldsymbol{x})=\frac{\boldsymbol{n}_k^T\boldsymbol{x}-d_k}{\boldsymbol{n}_k^T\boldsymbol{x}_k-d_k},\qquad k=0,1,\cdots,n. \tag{A.1}$$

Proof:

$$\boldsymbol{n}_k^T\boldsymbol{x}=\sum_{j=0}^{n}\mu_j\boldsymbol{n}_k^T\boldsymbol{x}_j=\mu_k\boldsymbol{n}_k^T\boldsymbol{x}_k+\sum_{\substack{j=0\\j\neq k}}^{n}\mu_j d_k$$
$$=\mu_k\boldsymbol{n}_k^T\boldsymbol{x}_k+(1-\mu_k)d_k.$$

Thus (A.1) follows, and we can immediately state the following proposition.

Proposition 2: For any $\boldsymbol{x}\in A$, if $\mu_k<0$, either

$$\boldsymbol{n}_k^T\boldsymbol{x}<d_k<\boldsymbol{n}_k^T\boldsymbol{x}_k \tag{A.2}$$

or

$$\boldsymbol{n}_k^T\boldsymbol{x}>d_k>\boldsymbol{n}_k^T\boldsymbol{x}_k$$

where $\boldsymbol{n}_k^T\boldsymbol{x}_k$ and $\boldsymbol{n}_k^T\boldsymbol{x}$ are the projections on $\boldsymbol{n}_k$ of the vectors $\boldsymbol{x}_k$ and $\boldsymbol{x}$, respectively. Geometrically, the inequalities in (A.2) imply that any line segment connecting $\boldsymbol{x}_k$ and $\boldsymbol{x}$ intersects the hyperplane H_k.

Next we consider the simplex $S(\boldsymbol{x}_0,\cdots,\boldsymbol{x}_n)$ in (1) and the boundary B_k in (2).

Proposition 3: Let $\boldsymbol{x}^i\in B_k$ and $\boldsymbol{x}(\epsilon)$ be given by (12), where ϵ is an arbitrarily small, positive number, then $\mu_k(\boldsymbol{x}(\epsilon))$ defined with respect to $\{\boldsymbol{x}_0,\cdots,\boldsymbol{x}_k,\cdots,\boldsymbol{x}_n\}$ is negative.

Proof: From (12)

$$\boldsymbol{n}_k^T\boldsymbol{x}(\epsilon)-d_k=d_k+\boldsymbol{n}_k^T\left(\frac{\epsilon}{2}\boldsymbol{x}_{k-1}-x_k+\frac{\epsilon}{2}\boldsymbol{x}_{k+1}\right)-d_k$$
$$=\epsilon\left(d_k-\boldsymbol{n}_k^T\boldsymbol{x}_k\right).$$

Thus $\mu_k(\boldsymbol{x}(\epsilon))=-\epsilon$ is negative.

From Proposition 2, we can say that the line segment joining $\boldsymbol{x}_k$ and $\boldsymbol{x}(\epsilon)$ intersects the hyperplane defined by the boundary B_k. And as $\epsilon\to 0$ $\boldsymbol{x}(\epsilon)\to\boldsymbol{x}^i$ which is on B_k.

Proposition 4: Let $\boldsymbol{x}_k=\boldsymbol{x}_{k-1}'+\boldsymbol{x}_{k+1}-\boldsymbol{x}_k$, then $\mu_k(\boldsymbol{x}_k')$ defined with respect to $\{\boldsymbol{x}_0,\cdots,\boldsymbol{x}_k,\cdots,\boldsymbol{x}_n\}$ is negative.

Proof:

$$\boldsymbol{n}_k^T\boldsymbol{x}_k'=2d_k-\boldsymbol{n}_k^T\boldsymbol{x}_k.$$

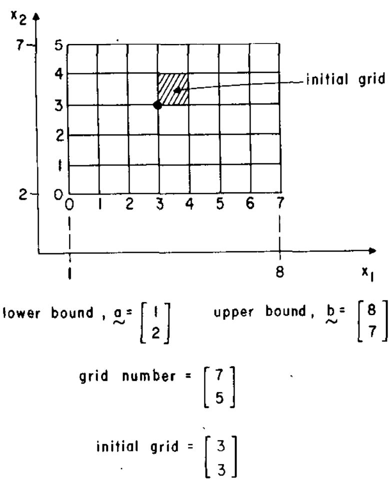
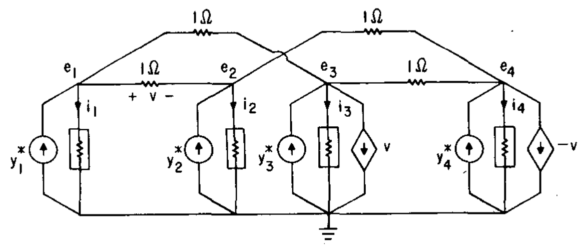


Fig. 12. Input data to program in 2-dimensional cases.



$$i_1 = 0.1e_1^3 - e_1^2 + e_1$$
$$i_2 = 0.1e_2^3 - e_2^2 + e_2$$
$$i_3 = e_3^3 + 2e_3^2 - 2e_3$$
$$i_4 = e_4^3 + 2e_4^2 - 2e_4$$
$$v = e_1 - e_2$$

Fig. 13. Circuit for Example 5.

Thus

$$\mu_k(x'_k) = \frac{n_k^T x'_k - d_k}{n_k^T x_k - d_k} = -1.$$

This implies that x'_k like $x(\epsilon)$ is on the opposite side of x_k with respect to the hyperplane defined by B_k. In (13) if ϵ is positive and sufficiently small we have a representation of $x(\epsilon)$ as an interior point of the new simplex $S(x_0, x_1, \cdots, x_{k-1}, x'_k, x_{k+1}, \cdots, x_n)$. Note all coefficients satisfy the condition $\mu_j \geq 0$ and $\sum_{j=0}^{n} \mu_j = 1$.

APPENDIX B

```
     INPUT Y            LOWER BOUND       UPPER BOUND     NO. OF GRID    INITIAL GRID   INITIAL POINT

 .2000E+01           -.1500E+02         .1500E+02          9              2             -8.3333
0.                   -.1500E+02         .1500E+02          9              2             -8.3333
 .2000E+01           -.1500E+02         .1500E+02          9              8             11.6667
0.                   -.1500E+02         .1500E+02          9              7              8.3333
      INITIAL SIMPLEX IS

   -8.33333  -5.00000  -5.00000  -5.00000  -5.00000
   -8.33333  -8.33333  -5.00000  -5.00000  -5.00000
   11.66667  11.66667  11.66667  15.00000  15.00000
    8.33333   8.33333   8.33333   8.33333  11.66667
   INITIAL  UI  IS

      .2000
      .2000
      .2000
      .2000
      .2000
      INITIAL  U  IS

      -.6016
       .0415
      2.4554
      -.3149
      -.5804

 SOLUTION IN SIMPLEX

         8.3333     8.3333     8.3333     5.0000     8.3333
         8.3333     8.3333     8.3333     5.0000     5.0000
         1.6667     5.0000     5.0000     1.6667     1.6667
         1.6667     1.6667     5.0000     1.6667     1.6667
      UI                 U

       .0000          .5941E+00
       .2749          .1168E-01
       .1591          .1029E-01
       .3384          .2409E+00
       .2277          .1430E+00
NO.   1 SOLUTION POINT IS                  CALCULATED INPUT IS           INPUT IS

                  .7530E+01                 -.2075E+00                  .2000E+01
                  .7054E+01                 -.2730E+01                 0.
                  .1740E+01                 -.7463E-02                  .2000E+01
                  .1701E+01                 -.9468E+00                 0.

  NO. OF SIMPLICES  CROSSED =        43
  THE SOLUTION CURVE REACHES LOWER BOUND          TIME REQUIRD  =      .404   SEC.
```

```
REFINE SEARCH FOR SOLUTION NO.    1

      INPUT Y              LOWER BOUND       UPPER BOUND       NO. OF GRID       INITIAL GRID      INITIAL POINT

   .2000E+01               .5000E+01         .8333E+01             9                 5                6.8519
  0.                       .5000E+01         .8333E+01             9                 5                6.8519
   .2000E+01               .1667E+01         .5000E+01             9                 5                3.5185
  0.                       .1667E+01         .5000E+01             9                 5                3.5185
        INITIAL SIMPLEX IS

     6.85185   7.22222   7.22222   7.22222   7.22222
     6.85185   6.85185   7.22222   7.22222   7.22222
     3.51852   3.51852   3.51852   3.88889   3.88889
     3.51852   3.51852   3.51852   3.51852   3.88889
     INITIAL  UI  IS

        .2000
        .2000
        .2000
        .2000
        .2000

        INITIAL  U  IS

        -4.1351
         1.1486
         6.3408
          .0962
        -2.4505

   SOLUTION IN SIMPLEX

            7.9630    7.9630    7.9630    7.9630    8.3333
            7.5926    7.5926    7.9630    7.9630    7.9630
            1.6667    2.0370    2.0370    2.0370    2.0370
            1.6667    1.6667    1.6667    2.0370    2.0370

        UI                  U

         .1672            .2011E+00
         .3446            .3097E+00
         .0000            .3390E-01
         .1795            .1439E+00
         .3087            .3114E+00
NO.    1      SOLUTION POINT IS                  CALCULATED INPUT IS              INPUT IS

                    .8078E+01                          .1958E+01                 .2000E+01
                    .7774E+01                        -.4569E-01                0.
                    .1963E+01                          .1875E+01                 .2000E+01
                    .1835E+01                        -.1884E+00                0.

     NO. OF SIMPLICES  CROSSED =        43                        TIME REQUIRED  =       .249   SEC.
```

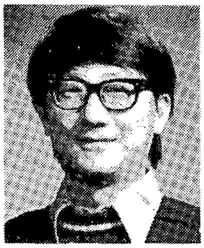
```
REFINE SEARCH FOR SOLUTION NO.    1

      INPUT Y              LOWER BOUND       UPPER BOUND       NO. OF GRID       INITIAL GRID      INITIAL POINT

   .2000E+01               .7963E+01         .8333E+01             9                 5                8.1687
  0.                       .7593E+01         .7963E+01             9                 5                7.7984
   .2000E+01               .1667E+01         .2037E+01             9                 5                1.8724
  0.                       .1667E+01         .2037E+01             9                 5                1.8724
        INITIAL SIMPLEX IS

     8.16872   8.20988   8.20988   8.20988   8.20988
     7.79835   7.79835   7.83951   7.83951   7.83951
     1.87243   1.87243   1.87243   1.91358   1.91358
     1.87243   1.87243   1.87243   1.87243   1.91358
     INITIAL  UI  IS

        .2000
        .2000
        .2000
        .2000
        .2000
        INITIAL  U  IS

         2.8815
        -1.6211
        -2.7980
         2.9977
         -.4601
   SOLUTION IN SIMPLEX

            8.0864    8.0864    8.1276    8.0864    8.0864
            7.7984    7.7984    7.7984    7.7572    7.7984
            1.9547    1.9959    1.9959    1.9547    1.9547
            1.8724    1.8724    1.8724    1.8313    1.8313

        UI                  U

         .2787            .4912E-01
         .0000            .4091E+00
         .2928            .5289E-01
         .2048            .2324E+00
         .2236            .2065E+00
NO.    1      SOLUTION POINT IS                  CALCULATED INPUT IS              INPUT IS

                    .8089E+01                          .2000E+01                 .2000E+01
                    .7787E+01                        -.4579E-03                0.
                    .1974E+01                          .1998E+01                 .2000E+01
                    .1852E+01                        -.2351E-02                0.

     NO. OF SIMPLICES  CROSSED =        21

        TIME REQUIRED  =       .167   SEC.
```

Acknowledgment

The authors would like to thank I. Cederbaum for his helpful comments.

References

[1] J. Katzenelson, "An algorithm for solving nonlinear resistive networks," *B.S.T.J.*, vol. 44, pp. 1605–1620, Oct. 1965.
[2] L. O. Chua, "Efficient computer algorithms for piecewise-linear analysis of resistive nonlinear networks," *IEEE Trans. Circuit Theory*, vol. CT-18, pp. 73–85, Jan. 1971.
[3] T. Ohtsuki and N. Yoshida, "DC analysis of nonlinear networks based on generalized piecewise-linear characterization," *IEEE Trans. Circuit Theory*, vol. CT-18, pp. 146–152, Jan. 1971.
[4] E. S. Kuh and I. N. Hajj, "Nonlinear circuit theory: resistive networks," *Proc. IEEE*, vol. 59, pp. 340–355, Mar. 1971.
[5] T. Fujisawa and E. S. Kuh, "Piecewise-linear theory of nonlinear networks," *SIAM J. Appl. Math.*, vol. 22, no. 2, pp. 307–328, Mar. 1972.
[6] T. Fujisawa, E. S. Kuh, and T. Ohtsuki, "A sparse matrix method for analysis of piecewise-linear resistive networks," *IEEE Trans. Circuit Theory*, vol. CT-19, no. 6, pp. 571–584, Nov. 1972.
[7] M. J. Chien and E. S. Kuh, "Solving piecewise-linear equations for resistive networks," *International Journal of Circuit Theory and Applications*, vol. 4, no. 1, pp. 3–24, Jan. 1976.
[8] M. J. Chien, "Piecewise-linear Theory and Computation of Nonlinear Resistive Networks," Ph.D. dissertation, Dep. of EECS, Univ. of California, Berkeley, Mar. 1975.
[9] T. E. Stern, *Piecewise-linear Network Theory*, MIT Tech. Rep. 315, 1956.
[10] M. Iri, "A method of multi-dimensional linear interpolation," (in Japanese), *J. Inform. Processing Soc. of Japan*, pp. 211–215, July 1967.
[11] J. M. Bennett, "Triangular Factors of Modified Matrices," *Numerische Mathematik*, vol. 7, pp. 217–221, 1965.
[12] Ortega and Rheinboldt, *Iterative Solution of Nonlinear Equations in Several variables*. New York: Academic Press, ch. 7.
[13] H. W. Kuhn, "Some combinatorial lemmas in topology," *IBM J. of Research and Development*, vol. 4, pp. 518–524, Nov. 1960.
[14] —— "Simplicial approximation of fixed points," in *Proc. of the National Academy of Sciences U.S.A.*, vol. 61, 1968, pp. 1238–1242.
[15] B. C. Eaves, *Solving Regular Piecewise-linear Convex Equations*, Department of Operations Research, Stanford University, CA, Aug. 1973.
[16] H. Scarf, *The Computation of Economic Equilibria*. New Haven and London, CN: Yale University Press, 1973, ch. 7, pp. 37–50.
[17] C. E. Lemke, "Bimatrix equilibrium points and mathematical programming," *Management Science*, vol. 11, pp. 681–689, 1965.
[18] H. Scarf, "The approximation of fixed points of a continuous mapping," *SIAM J. of Applied Mathematics*, vol. 15, pp. 1328–1343, 1967.
[19] B. C. Eaves, "Computing Kakutani fixed points," *SIAM J. of Appl. Math.*, vol. 21, no. 2, pp. 236–244, 1971.
[20] I. W. Sandberg and A. N. Willson, Jr., "Existence of solutions for the equations of transistor-resistor-voltage source networks," *IEEE Trans. Circuit Theory*, vol. CT-18, pp. 619–625, Nov. 1971.
[21] F. F. Wu, "Existence of an operating point for a nonlinear circuit using the degree of mapping," *IEEE Trans. on Circuits and Systems*, vol. CAS-21, pp. 671–677, Sept. 1974.
[22] K. M. Adams, "Bounds on the branch voltages of nonlinear resistive networks," in *1974 European Conference on Circuit Theory and Design*.
[23] A. N. Willson, Jr., "Three-terminal no-gain elements," in *Proc. 1975 International Symposium on Circuits and Systems*, pp. 29–32, 1975.
[24] M. T. Wu, "Solving nonlinear equations using piecewise-linear analysis and simplicial subdivision," Master's project report, Department of Electrical Engineering and Computer Sciences, University of California, Berkeley, CA, June 1976.



Ming-Jeh Chien (S'71–M'75) was born on August 22, 1947 in Taiwan. He was educated in Taiwan and received B.S.E.E. degree in 1969 from National Chiao-Tung University, Taiwan. He began research work under Dr. Ernest S. Kuh in January 1971, and received the M.S. (June 1971) and Ph.D. (1975) degrees, both in the field of electrical engineering from the University of California, Berkeley.

Since March, 1975, he has been an Assistant Professor of Electrical and Computer Engineering, Wayne State University, Detroit, MI. His research interests are circuit modeling and analysis, economic system modeling and analysis, numerical analysis, mathematical programming, and computer-aided circuit analysis.

Ernest S. Kuh (S'49–A'52–M'57–F'65) received the B.S. degree from the University of Michigan, Ann Arbor, in 1949, the M.S. degree from the Massachusetts Institute of Technology, Cambridge, in 1950, and the Ph.D. degree from Stanford University, Stanford, CA, in 1952.

From 1952 to 1956, he was a member of the Technical Staff at the Bell Telephone Laboratories, Murray Hill, NJ. He joined the Electrical Engineering Department faculty at the University of California, Berkeley, in 1956. From 1968 to 1972 he served as Chairman of the Department of Electrical Engineering and Computer Sciences. Since September, 1973, he has served as Dean, College of Engineering, at the University of California, Berkeley. He is co-author of three books: *Principles of Circuit Synthesis*, 1959; *Theory of Linear Active Networks*, 1967; and *Basic Circuit Theory*, 1969.

Dr. Kuh was President of the IEEE Circuits and Systems Society in 1972. Since January 1, 1976, he has been a member of the Board of Directors of IEEE. At present, he serves on the Visiting Committee of the General Motors Institute. He is a member of the NSF Advisory Panel on Electrical Sciences and Analysis. He also serves as a member of the Evaluation Panel for the Institute for Applied Technology of the National Bureau of Standard. Among the various honors and awards he has received are as follows: NSF Senior Postdoctoral Fellow (1962), Miller Research Professorship, Berkeley (1965), National Electronics Conference Award (1966), Distinguished Alumni Award, University of Michigan (1970), IEEE Guillemin–Cauer Award (1973), and membership in the National Academy of Engineering (1975).