数理解析研究所講究録
第787巻 1992年 163-175

簡単な解曲線追跡法

Simple Algorithms for Tracing Solution Curves

山村清隆 (Kiyotaka Yamamura)

群馬大学工学部情報工学科

Faculty of Engineering, Gunma University

Abstract — This paper presents some simple and practical algorithms for tracing implicitly defined solution curves. These algorithms use hyperspheres instead of hyperplanes which are used in the typical predictor-corrector algorithms. Effective techniques for preventing the "reversion" phenomenon of the curve tracing are also proposed. The proposed algorithms are geometrically clear and can be easily programmed.

I. INTRODUCTION

This paper deals with a system of n nonlinear equations in n+1 variables

$$\begin{aligned} &f_1(x_1,x_2,\cdots,x_n,x_{n+1})=0\\ &f_2(x_1,x_2,\cdots,x_n,x_{n+1})=0\\ &\qquad\vdots\\ &f_n(x_1,x_2,\cdots,x_n,x_{n+1})=0. \end{aligned} \tag{1}$$

Since there is one more variable than there are independent equations, the set of all $\mathbf{x}=(x_1,x_2,\cdots,x_n,x_{n+1})^T$ that satisfy (1) will generally consist of one or more solution curves in the (n+1)-dimensional Euclidean space.

To find all solutions of (1) is a basic problem which is widely encountered in science and engineering. For example, to obtain the driving-point characteristics and transfer characteristics of nonlinear resistive circuits, it is necessary to solve an equation of the form (1), where x_{n+1} denotes either the driving-point voltage or current.

Another common situation which calls for the solution of (1) is the application of t e homotopy method (or the continuation method) for solving a system of n nonlinear equations in n variables

$$\mathbf{f}(\mathbf{x})=\mathbf{0}, \tag{2}$$

where $\mathbf{f}: \mathbf{R}^n \to \mathbf{R}^n$ is a C^2 mapping and $\mathbf{x}\in\mathbf{R}^n$. In the homotopy method, we introduce an extra variable $x_{n+1}\triangleq t$, in addition to $\mathbf{x}=(x_1,x_2,\cdots,x_n)^T$, and define a new system of n equations

$$\mathbf{h}(\mathbf{x},t)=\mathbf{0} \tag{3}$$

in n+1 variables. The mapping **h** is called a homotopy. One of the most common homotopies is the Newton homotopy

$$\mathbf{h}(\mathbf{x},t)=\mathbf{f}(\mathbf{x})-(1-t)\mathbf{f}(\mathbf{a}), \tag{4}$$

where **a** is any initial guess. At t=0, **a** is a solution of (3), and at t=1, (3) reduces to (2). Hence, by tracing the solution curve starting from (**a**,0), we can obtain the solutions of (2) at t=1. This homotopy method is known to be globally convergent for many practical problems including nonlinear circuit analysis.

Many algorithms have been presented for tracing the solution curves [1]-[16]. These curve tracing algorithms are roughly divided into two categories; one is the predictor-corrector algorithms (or the differential algorithms) and the other is the piecewise-linear algorithms. These algorithms are very efficient at least for small problems. Especially, the backward-differentiation formula curve tracing algorithm (which is called the BDF algorithm in this paper) proposed by Ushida and Chua [12] is one of the most effective and reliable algorithms for tracing solution curves which have sharp turning points.

However, these curve-tracing algorithms are not widely used in practical applications. One of the reasons is that the theory and the programming of these algorithms are not so easy for the practical engineers and scientists who are not familiar with the homotopy methods. Especially in the computer-aided design of electronic circuits, the circuit simulation depends largely on the existing circuit simulators, and it is difficult to make a new program of dc analysis from the first.

In this paper, we propose some simple algorithms for tracing solution curves. These algorithms are geometrically clear and can be easily programmed, therefore they will be widely used by many engineers. Another advantage of the algorithms is that the "reversion" phenomenon of the curve tracing (which often occurs in the BDF algorithm) is prevented by special techniques. It is also shown that the algorithms can be easily implemented on the existing circuit simulators such as SPICE.

II. THE SPHERICAL ALGORITHM

The typical predictor-corrector algorithms use tangents as predictors and hyperplanes as correctors as shown in Fig.1 [1],[2]. In the predictor step, the next point is predicted to obtain a good starting approximation of Newton's method. In the corrector step, a system of nonlinear equations is solved by Newton's method to get back onto the solution curve. The

stepsize of the predictor is pertinently determined; for example, if Newton's method does not converge or seems to require many iterations, the stepsize is set small. Also, if the number of Newton iterations is small, the stepsize is set large. In the BDF algorithm, an efficient stepsize control algorithm is utilized [12].

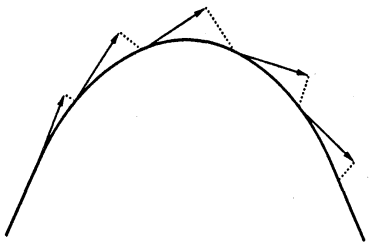


Fig.1

However, the predictor-corrector algorithms using the hyperplanes are not simple and their programming is complicated. Another demerit of these algorithms is that they become extremely slow in the neighbourhood of sharp turning points of the solution curves. In Fig.2, the stepsize of the predictor is made smaller and smaller until the hyperplane intersects the solution curve.

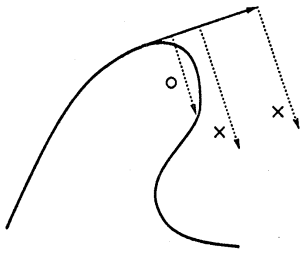


Fig.2

In this section, we propose a new algorithm which uses hyperspheres instead of hyperplanes. As shown in Fig.3, a sphere which encloses a part of a solution curve always intersects the solution curve at least two times. Hence, the corrector equation has at least two solutions.

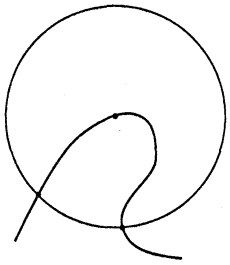


Fig.3

Let $\mathbf{x}^0$ be a point on the solution curve. Then the equation describing the (n+1)-dimensional sphere centred at $\mathbf{x}^0$ with radius s is given by

$$(x_1-x_1^0)^2+(x_2-x_2^0)^2+\cdots+(x_{n+1}-x_{n+1}^0)^2=s^2. \tag{5}$$

Therefore the intersections of the sphere and the solution curve (A and B in Fig.4) are the solutions of the following system of n+1 equations

$$\begin{aligned} &f_1(x_1,x_2,\cdots,x_n,x_{n+1})=0\\ &f_2(x_1,x_2,\cdots,x_n,x_{n+1})=0\\ &\qquad\vdots\\ &f_n(x_1,x_2,\cdots,x_n,x_{n+1})=0.\\ &(x_1-x_1^0)^2+(x_2-x_2^0)^2+\cdots+(x_{n+1}-x_{n+1}^0)^2=s^2. \end{aligned} \tag{6}$$

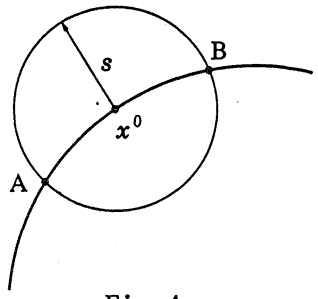


Fig.4

As shown in Fig.5, the solution curve can be traced by successively solving the systems of nonlinear equations (6). In each step, the previous solution is used as the centre of the new sphere. The radius of the sphere is made small if Newton's method does not seem converge in the corrector step. The initial radius is determined by the stepsize control algorithm in [12].

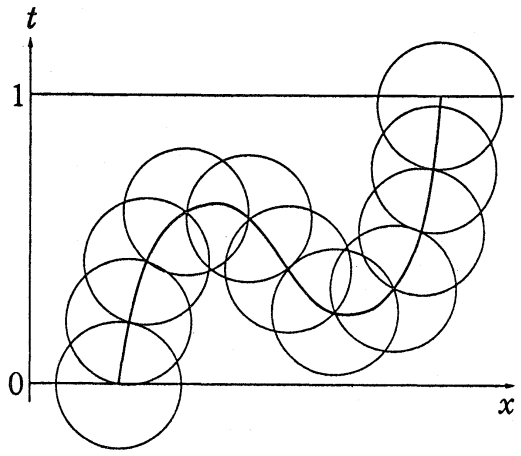


Fig.5

Each corrector equation has at least two solution, one of which lies in the forward direction and another of which lies in the backward direction. In order to make the Newton iteration converge to the desired solution in the forward direction, the predictor algorithm is performed. There are many predictor algorithms, the most efficient one of which is the BDF predictor algorithm proposed by Ushida and Chua [12]. In order to make the algorithm simple, we consider to use the first order BDF predictor algorithm. Then, as shown in Fig.6, the predicted value is obtained by extrapolating the two previous solutions which are on the solution curve.

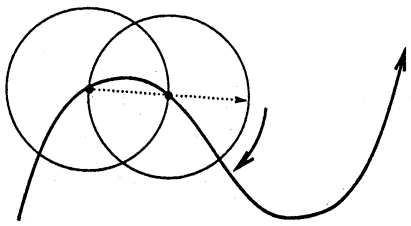


Fig.6

This is the basic form of the spherical algorithm proposed in this paper. This algorithm is simple and geometrically clear, so it may be widely used by practical designers.

Remark 1: The spherical algorithm proposed in this section is not a new algorithm because it is essentially equivalent to the first order BDF algorithm. Mr. Yasuaki Inoue of Sanyo Electric Co., Ltd. was the first to point this out [16]. However, the concept "sphere" makes the algorithm geometrically very clear and gives considerable insight to the algorithm. As shown in later sections, this geometrical interpretation makes it easy to detect the potential difficulties of the BDF algorithm and to find techniques for overcoming them. □

III. JACOBIAN MATRIX

The advantages of the spherical algorithm are its simplicity and ease of programming. In circuit simulation, there is another advantage.

Existing circuit simulators such as SPICE use Newton's method or its variants to solve nonlinear equations. However Newton's method is not globally convergent and requires a good initial guess for convergence. Guessing a solution is often a difficult task, and many circuit designers experience difficulties in finding dc solutions using the circuit simulators. The homotopy methods are known to be globally convergent for

nonlinear circuit equations, but they have not been widely used because the present circuit simulation depends largely on the existing circuit simulators. Hence, it is desirable to improve the convergence through a little modification of the existing circuit simulator programs.

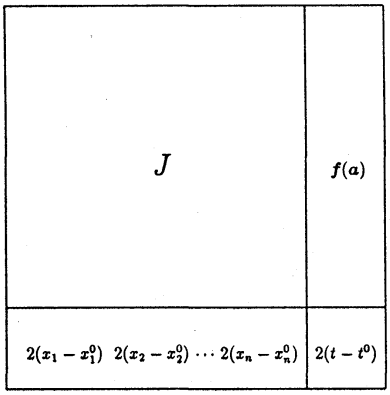
Let us consider a system of nonlinear equations (2) which describes a nonlinear resistive circuit. We shall solve (2) by the homotopy method using the Newton homotopy

$$\mathbf{h}(\mathbf{x},t)=\mathbf{f}(\mathbf{x})-(1-t)\mathbf{f}(\mathbf{a}). \tag{7}$$

Then the corrector equation in the spherical algorithm is

$$\mathbf{f}(\mathbf{x})-(1-t)\mathbf{f}(\mathbf{a})=\mathbf{0}$$
$$(x_1-x_1^0)^2+(x_2-x_2^0)^2+\cdots+(x_n-x_n^0)^2+(t-t^0)^2=s^2. \tag{8}$$

The Jacobian matrix of this equation is



where J is the Jacobian matrix of $\mathbf{f}$. This (n+1)x(n+1) Jacobian matrix is obtained by adding one row and one column to the nxn matrix J.

In the existing circuit simulators such as SPICE, the program of Newton's method for $\mathbf{f}(\mathbf{x})=\mathbf{0}$ is already described. Therefore the spherical algorithm is easily programmed through a little modification of the existing program. Thus, the spherical algorithm can be readily implemented on the circuit simulators.

IV. THE IMPROVED SPHERICAL ALGORITHMS

Let us consider Fig.6 again. If the sphere is sufficiently small, Newton's method will converge to the desired solution. However, using small spheres is time consuming, and it is desirable to use large spheres to make the algorithm efficient. In such a case, Newton's method may not converge to the desired solution. Instead, it may converge to the solution

which lies in the backward direction as shown in Fig.7. Such a phenomenon is called the "reversion" of the curve tracing.

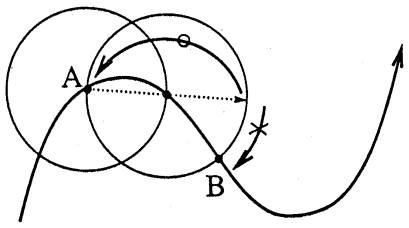


Fig.7

If the radius of the new sphere is the same as the previous one, this reversion can be detected because the unwanted solution A coincides with the previous solution. However, if the radius changes, the reversion cannot be detected because the solution A does not coincide with the previous solution as shown in Fig.8. In this case, the curve tracing may reverse from this point. In the BDF algorithm, this reversion phenomenon often occured in practical applications. In this section, we propose two improved algorithms which overcome this problem.

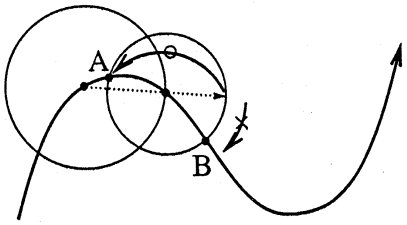


Fig.8

In the first improved algorithm, we move the centre of the sphere a little so that the new sphere passes the previous solution point. In Fig.9, point A is the previous solution, point D is the solution which has just been obtained, point C is the centre of the new sphere, and point B is the next solution which we are going to seek. The centre C is determined by the first or second order interpolation or extrapolation.

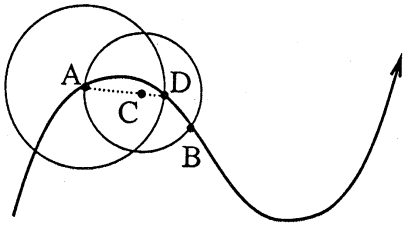


Fig.9

Fig.10 describes how to reduce the radius of the sphere when Newton's method does not converge in the corrector step. For simplicity, we will halve the step size of the predictor if convergence does not occur. Suppose that we first use a sphere which is the same size as the previous one and Newton's method does not converge to the solution B. In order to halve the stepsize of the predictor, we use a sphere with radius 3/4 times as large as the previous one. If convergence does not occur again (for example, Newton's method converges to the previous solution A), we reduce the radius 5/6 times to make the stepsize 1/2. Repeating this process, the predicted point approaches the desired solution, and convergence will occur in a finite number of steps. Notice that all spheres pass the point A, so that the reversion can be detected.

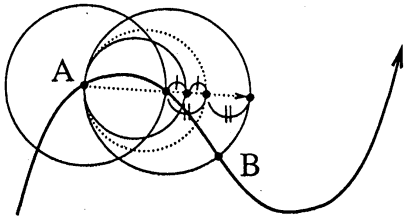


Fig.10

Next, we shall propose the second improved algorithm. In this algorithm, we use a sphere passing the solution point which has just been computed. Fig.11 illustrates this algorithm. The centre of the sphere is always determined by (the first or second order) extrapolation. In this case, we can also detect the reversion.

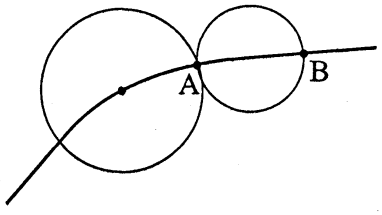
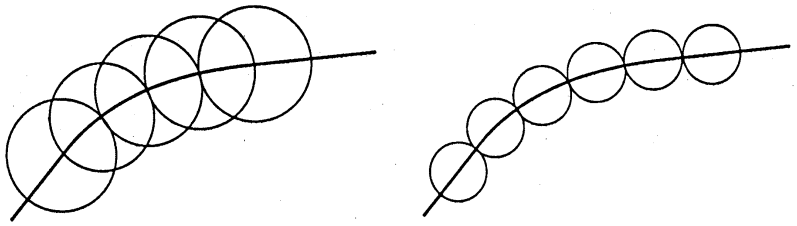


Fig.11

In the basic form of the spherical algorithm, the radius of the spheres determines how far the algorithm moves in one step. In the second improved algorithm, the diameter determines the stepsize as shown in Fig.12.

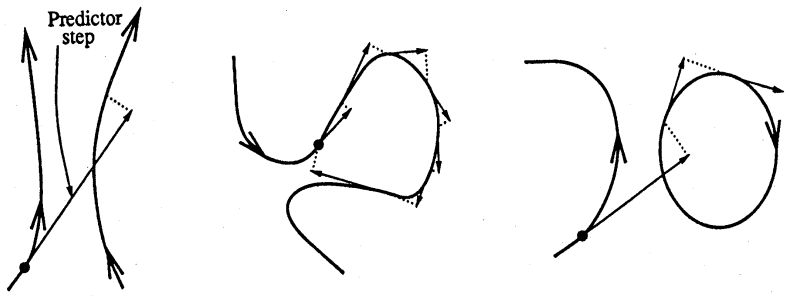


(a) (b)

Fig.12

Now we shall compare the first and the second improved algorithms. As shown in Fig.12, the second algorithm uses smaller spheres if the stepsize of the predictor is the same.

In all of the predictor-corrector algorithms, the following potential dangers can occur; the algorithm goes to a different solution curve; it cycles; it attaches itself to a loop as shown in Fig.13.



(a) (b) (c)

Fig.13

As shown in Fig.14, the second algorithm uses smaller spheres and this reduces the possibility that those dangers occur.

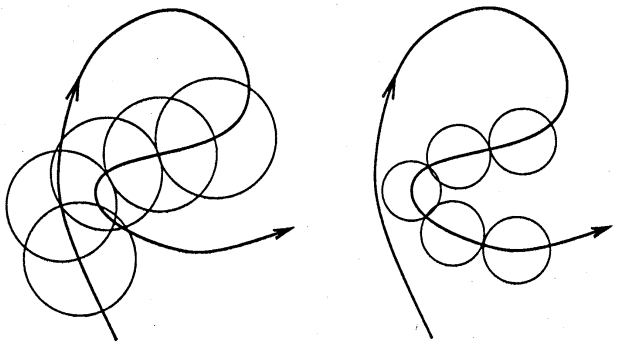


Fig.14

However, in the second algorithm, the centre of the sphere is always determined by xtrapolation. As shown in Fig.15, the centre may be far away from the solution curve in the vicinity of sharp turning points. Therefore, the second algorithm will require a good predictor.

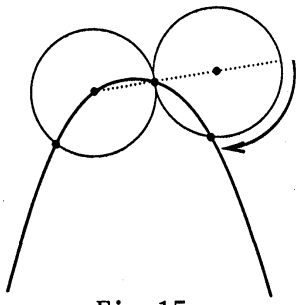


Fig.15

V. THE ELLIPTIC ALGORITHM

In this section, we shall consider an algorithm which uses ellipses. Fig.16 illustrates the algorithm. The use of ellipses which are long in suitable directions (for example, the direction of the variable which exhibits the maximum variation in the predictor step) is often more effective than the spherical algorithms.

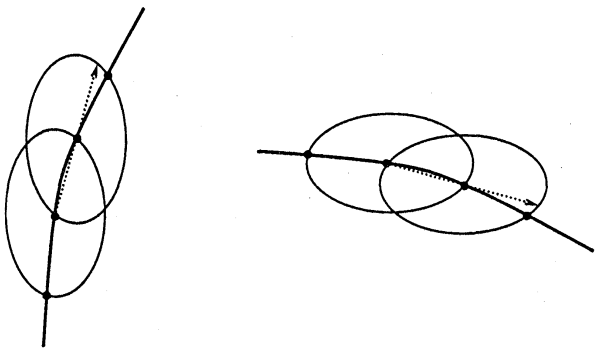


Fig.16

The elliptic algorithm is not a new algorithm because it is a variant of the first order BDF algorithm. Mr. Yasuaki Inoue of Sanyo Electric Co., Ltd. also condidered the use of ellipses. In this section, we show that it is possible to prevent the reversion phenomenon in the elliptic algorithm.

Consider an (n+1)-dimensional ellipse centred at $\mathbf{x}^0$ with radii $s, \lambda_2 s, \cdots, \lambda_{n+1} s$ (the radius in the x_i direction is $\lambda_i s$). Then the equation describing the ellipse is

$$\left(\frac{x_1-x_1^0}{s}\right)^2+\left(\frac{x_2-x_2^0}{\lambda_2 s}\right)^2+\cdots+\left(\frac{x_{n+1}-x_{n+1}^0}{\lambda_{n+1} s}\right)^2=1. \tag{9}$$

Let $\mathbf{x}^p$ be the previously obtained solution. Given points $\mathbf{x}^0$, $\mathbf{x}^p$ and the ratio $\lambda_2, \cdots, \lambda_{n+1}$, let us now determine the the ellipse which passes the point $\mathbf{x}^p$. Substituting $\mathbf{x}^p$ to Eq.(9), we have

$$\left(\frac{x_1^p - x_1^0}{s}\right)^2 + \left(\frac{x_2^p - x_2^0}{\lambda_2 s}\right)^2 + \cdots + \left(\frac{x_{n+1}^p - x_{n+1}^0}{\lambda_{n+1} s}\right)^2 = 1 \tag{10}$$

and

$$s^2 = (x_1^p - x_1^0)^2 + \left(\frac{x_2^p - x_2^0}{\lambda_2}\right)^2 + \cdots + \left(\frac{x_{n+1}^p - x_{n+1}^0}{\lambda_{n+1}}\right)^2 . \tag{11}$$

Therefore all radii can be determined from (11), and it is possible to prevent the reversion phenomenon also in the elliptic algorithm .

VI. NUMERICAL EXAMPLE

We have applied the homotopy methods using the spherical algorithms to various types of nonlinear resistive circuits including the simple multi-tunnel diode circuit and the four-transistor multi-state circuit discussed in [13].

When we applied the basic spherical algorithm (which is equivalent to the first order BDF algorithm) to these circuits, the reversion phenomenon sometimes occured. In these cases, the curve tracing went back to t=0 and solutions could not be obtained. However, when we applied the improved spherical algorithms, the reversion occured but it was always detected, and the solution was obtained in most of the examples. In a few examples, the phenomena described in Fig.13 occured and solution could not be obtained.

The computational efficiency of the algorithms depended largely on the parameters such as the starting point, the initial stepsize, and the maximum stepsize. It was not clear which algorithm was the most efficient and it seemed to be case by case.

We shall show one experimental result. The example circuit is the four-transistor multi-state circuit in [13] which is described by a system of 8 equations. We used the Newton homotopy as a homotopy, and used the following parameters; the starting point=**0**; the initial stepsize=0.05; the maximum stepsize=1; the order of the predictor=1; and the maximum number of Newton iterations in one trial=20. Then the total numbers of Newton iterations were 101 in the first algorithm and 108 in the second algorithm. These numbers include the numbers for divergence. The numbers of spheres we used were 18 and 19, respectively.

VII. CONCLUSION

In this paper, we have proposed some simple algorithms for tracing the solution curves. The spherical algorithms are geometrically clear and can be easily programmed, so they will be widely used by many engineers. Also, the proposed spherical algorithms can prevent the reversion of the curve tracing. The next subject we should consider is to develop a technique which prevent the phenomena described in Fig.13. Perhaps the spherical interpretation makes it easy to find such a technique.

ACKNOWLEDGMENT

The author would like to thank Mr. Yasuaki Inoue of Sanyo Electric Co., Ltd., Gunma, Japan, and Prof. Akio Ushida of Tokushima University, Tokushima, Japan, for their fruitful discussions and comments. The author also acknowledges Prof. Kazuo Horiuchi and Prof. Shin'ichi Oishi of Waseda University, Tokyo, Japan, for their constant encouragement. The author is also grateful to Mr. Hiroshi Yasuda of Hitachi Ltd., Gunma, Japan, and Mr. Gen Ushiki of Yamamura Laboratory for their help in the numerical experiments.

REFERENCES

[1] C.B.Garcia and W.I.Zangwill, Pathways to Solutions, Fixed Points, and Equilibria, Englewood Cliffs, NJ: Prentice-Hall, 1981.

[2] E.Allgower and K.Georg,"Simplicial and continuation methods for approximating fixed points and solutions to systems of equations," SIAM Review, vol.22, no.1, pp.22-84, Jan.1980.

[3] D.F.Davidenko, "On the approximate solution of a system of nonlinear equations," Ukraine Math. Zurnal, vol.5, pp.196-206, 1953.

[4] Y.Shinohara, "A geometric method of numerical solution of nonlinear equation and error estimation by Urabe's proposition," RIMS, Kyoto University, vol.5, pp.1-9, 1969.

[5] F.H.Branin, "Widely convergent method for finding multiple solutions of simultaneous nonlinear equations," IBM J. Res. Develop., vol.16, pp.504-522, 1972.

[6] K.S.Chao, D.K.Liu and C.T.Pan, "A systematic search method for obtaining multiple solutions of simultaneous nonlinear equations", IEEE Trans. Circuits Syst., vol.CAS-22, no.9, pp.748-753, Sept.1975.

[7] E.Ikeno and A.Ushida, "The arc-length method for the computation of characteristic curves," IEEE Trans. Circuits Syst., vol.CAS-23, no.3, pp.181-183, March 1976.

[8] L.O.Chua and A.Ushida, "A switching-parameter algorithm for finding multiple solutions of nonlinear resistive networks," Int. J. Cir. Theor. Appl., vol.4, no.3, pp.215-237, July 1976.

[9] T.Ohtsuki, T.Fujisawa, and S.Kumagai, "Existence theorems and a solution algorithm for piecewise-linear resistor networks," SIAM J. Math. Anal., vol.8, no.1, pp.69-99, Feb.1977.

[10] M.J.Chien and E.S.Kuh,"Solving nonlinear resistive networks using piecewise-linear analysis and simplicial subdivision," IEEE Trans. Circuits Syst., vol.CAS-24, no.6, pp.305-317, June 1977.

[11] K.S.Chao and R.Saeks, "Continuation methods in circuit analysis," Proc. IEEE, vol.65, pp.1187-1194, Aug.1977.

[12] A.Ushida and L.O.Chua, "Tracing solution curves of non-linear equations with sharp turning points," Int. J. Cir. Theor. Appl., vol.12, no.1, pp.1-21, Jan.1984.

[13] K.Yamamura and K.Horiuchi, "A globally and quadratically convergent algorithm for solving nonlinear resistive networks," IEEE Trans.Comput.-Aided Des. Integrated Cirsuits Syst., vol.9, no.5, pp.487-499, May 1990.

[14] L.Vandenberghe and J.Vandewalle, "Variable dimension algorithm for solving resistive circuits," Int. J. Cir. Theor. Appl., vol.18, no.5, pp.443-474, sept.1990.

[15] L.Trajkovic, R.C.Melville, and S.C.Fang, "Finding DC operating points of transistor circuits using homotopy methods," in Prof. IEEE Int. Symp. on Circuits and Systems, Singapore, pp.758-761, June 1991.

[16] Y.Inoue, "DC analysis of non-linear circuits using solution-tracing circuits," Trans. IEICE, vol.J74-A, 1991.