Modified method of hyperspheres: tracing homotopic bounded paths in nonlinear circuits

by

Delia Torres Muñoz

A dissertation submitted to the Electronics Department in partial fulfillment of the requirements for the degree of

D.Sc. in Electronics

at the

National Institute for Astrophysics, Optics and Electronics

August 2015

Tonantzintla, Puebla

Advisor:

Dr. Luis Hernández Martínez -INAOE
Dr. Héctor Vázquez Leal -UV

Resumen

Las ecuaciones que describen el comportamiento de un circuito electrónicos son no lineales debido a circuitos con elementos no lineales, y el análisis en CD es el primer paso que se sigue para el análisis de un circuito no lineal y encontrar la solución o soluciones del sistema de ecuaciones. Los métodos basados en el algoritmo de Newton Raphson presentan desventajas para resolver problemas no lineales, además no son capaces de hallar múltiples puntos de operación. Para mejorar los problemas anteriones los métodos de homotopía pueden hallar múltiples soluciones para un sistema de ecuaciones; sin embargo la convergencia de los métodos de homotopía dependen del punto de inicio, de los métodos de trazado y de no-linealidades características del sistema a resolver. La formulación de la homotopía requiere de técnicas adecuadas para el trazado de la curva; el algoritmo de trazado usado en este trabajo es el método hiperesférico el cual fue elegido porque es geométricamente claro lo cual facilita su programación. El método hiperesférico traza las trayectorias usando esferas de radio fijo donde cada esfera cruza a la curva al menos en dos puntos. No obstante al implementar en programación el método hiperesférico presenta algunos problemas como lo es la reversión de la curva y la lentitud en el trazado de la misma. En esta tesis se proponen dos metodologías para resolver problemas como la reversión y agilizar el trazado de la curva usando menos iteraciones. La primera metodología propuesta se basa en el cálculo del vector normal a la curva para detectar el problema de reversión además de una metodología para evitar la reversión. La segunda metodología logra reducir el número de iteraciones y el tiempo de cómputo durante el proceso de trazado de la curva. Ambas estrategias son implementadas y programadas para el método de hiperesferas, y son aplicadas para hallar la solución de diferentes casos de estudio obteniendo resultados satisfactorios durante el trazado de la curva.

Abstract

The equations describing the behavior of a electronic circuit are nonlinear due to nonlinear elements and the DC analysis is the first step for analyzing a nonlinear circuit and also to find the solution, or solutions, of the system of equations. Methods like Newton Raphson algorithm present disadvantages to solve nonlinear problems besides are not capable to find multiple solutions. To overcome above problems the homotopy methods can locate multiple solutions for a system of equations; however homotopy methods convergence depends of the starting point, the continuous methods and nonlinearities. The homotopy formulation require suitable path tracking techniques to accurately trace the homotopy curve, the algorithm to path-tracking used in this work is the spherical algorithm because it is geometrically clear and this characteristic can facilitate its programming. However during programming hyperspheres method there were some problems as reversion of the path on the curve and slow for tracing. In this thesis, two proposed methodologies are used to solve the above problems, the first proposed methodology is based in calculating the normal vector to the curve for detecting the problem of reversion then the methodology is used to avoid the problem of reversion. The second methodology is achieved during the tracing of the homotopy curve to reduce the number of iterations and the computational time. Both strategies are implemented and programmed for the hyperspheres method; also several case studies are solved, and we found satisfactory results for the path tracking problem.

Contents

Contents

List of Figures

List of Figures

Chapter 1

Introduction

Finding the DC operating points is a essential task in electrical circuit simulation that involves solving systems of non-linear algebraic equations (NAES). The NAES describe the behavior of semiconductor devices employing nonlinear elements [10]. Traditional simulation methods for solving nonlinear equations frequently exhibit difficulties of convergence. This problem is tackled in the circuital simulators by using Newton-Rapshon (NR) method and modified or damped NR [10, 11].

1.1   DC Analysis

The technologies are moved closer to their limits of performance where nonlinear effects will assume greater importance. An important objective in the analysis of an electronic circuit is to find its quiescent or DC operating point. Furthermore, the impact of the computers on circuit analysis and circuit design as well as the advent of new electronic devices and integrated circuits have generated much interest in nonlinear circuit theory [5].

Such an analysis is necessary to establish the correct DC-bias conditions of the circuit as required by the intended application. For example, the DC operating point is used as the starting point for transient analysis (circuit response in the time domain). Circuit design algorithms also need the dc operating point of the circuit. In this case, the operating point is required to evaluate the DC performance of the current design under a given set of constraints on the circuit components. In Most practical electronic circuits a constant steady state resulting from D.C excitations is the most commonly used simulation mode. If system performance is based on variations of the instantaneous signal variations then there is a reference signal impressed by the energy source. In mode analysis of small signal operation, signals vary slightly in the vicinity of the bias. Therefore if small incremental

signal are presented, the behavior of nonlinear circuit depends not only on its topology and the character of branches but also on the bias impressed. The design process for analog integrated circuits always start the D.C analysis and verification of D.C signals.

1.2 Analysis of Nonlinear Circuits

Nonlinear circuits differ from linear circuits in that they may have no solution, an infinite number of solutions, a unique solution or several solutions. All electronic circuits are nonlinear, the defining relationships between the circuit variables (usually voltages and currents) are nonlinear [9]. During the analysis of electronic circuits some electronic elements are approximately linear in a relatively large operating region. Nonlinearities are undesired in many applications such as filters and electronic amplifiers, while in other applications as modulation, demodulation, coding and decoding the nonlinear effects are essential for the suitable application. A very important class of non-linear applications is transistor circuit for building logic circuits used in digital computers, unlike for the linear circuits, there is no analytical method to solve nonlinear circuits. The nonlinearities stem from nonlinear device models, which can contain polynomial or exponential terms, and may not be smooth. Circuit simulators developed in the 1970 are characterized by using techniques for Newton-Raphson (NR) or one of its variants for iteration and implicit integration of nonlinear differential equations and the use of transistor models. Design engineers particularly in analog circuit design spend much time using the simulator to find an operating point. In some cases, the difficulty of obtaining an operating point signals that something is wrong with the circuit and indicates that a redesign is needed. In others cases the difficulty is simply an artifact of the simulators algorithm used to find and operating point. For such a case the designers time spent finding a operating point has contributed nothing to the design process [7]. Circuits that contain nonlinear elements may have multiple discrete DC operating points. On the other hand circuits consisting of positive linear resistor posses either one DC operating point, or, in special cases, a continuous family of DC operating points. The interest in nonlinear circuit theory can be attributed to two factors: the impact of the computers on circuit analysis and design, and the advent of new devices and integrated circuits. In the analysis of nonlinear networks, the first step is often the determination of the equilibrium points or operating points. This problem amount to solving the nonlinear algebraic equation $f(x, u) = 0$ from the given set of state equations $\dot{x} = f(x, u)$ describing the network (where x is output, u is input and $\dot{x}$ is state variable), which is equivalent to finding solutions of the nonlinear resistive network model obtained by shorting out all inductors and opening all capacitors in the

original network.

1.2.1 Analog circuit containing diodes and transistors

The evaluation of the DC quiescent operating points of an analog electrical circuits have three main purposes:

1. The verification of the correct DC values in the circuit.

2. The computation of the initial conditions for the nonlinear transient analysis.

3. The evaluation of the proper parameter values of the linearized, small-signal models for the nonlinear devices, which are needed for linear AC analysis.

Commonly the Newton Raphson method is applied with some of its modifications for DC analysis. Starting from a initial guess of the solution, we obtain iteratively improved estimates of the actual solution until some convergence criterion is satisfied. Circuits containing nonlinear elements may have multiple discrete DC operating points. Such circuits consist of exponential diodes, Ebers-Moll modeled bipolar transistors and insulated-gate field-effect transistors (FETs). A diode will be modeled by exponential function. Whereas that a bipolar NPN transistor will be represented by Ebers-Moll model (See Figure 1.1).
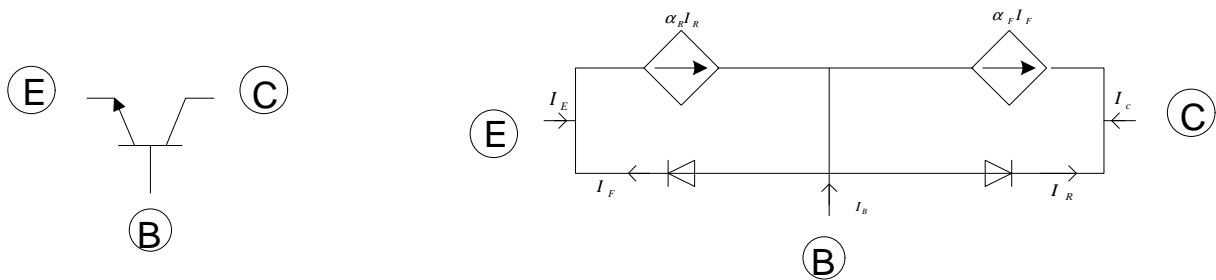
Figure 1.1: Ebers-Moll model of a NPN transistor

This model consists of two exponential diodes, and two current transactors whose currents are controlled by diode currents. After selecting a suitable mathematical model for the electrical element then the equations describing the behavior of the circuit are obtained; a circuits operating points are solutions of a system of nonlinear equations of the form [6]:

$$F(x) = 0. \tag{1.1}$$

where F represents the nonlinear expressions and x is the system variables.

5

1.2.2   Number of DC operating points

Nonlinear circuits consisting of an arbitrary number of linear resistors and diodes posses at most one DC operating point. Several fundamental result relate the topology of a transistor circuit to the number of possible DC operating points. In general, any circuit that does not posses a unique DC operating. One circuit that contains more than two transistors may posses numerous operating points. Several methods has been proposed to obtain upper bounds on the number of DC operating points of transistor circuits.

1.2.3   Iterative method for analysis of nonlinear circuits

The most significant feature of nonlinear analysis is the use of iterative methods. DC analysis belongs to nonlinear resistive networks containing diodes and transistors. For a transistor circuit, the DC analysis is realized by treating capacitors as open circuits an inductors as short circuit. For time domain analysis is based on the choice of discrete time instants t_n, for $n = 1, 2, 3....$; where the input exitation may be an arbitrary function of time. For solving the non-linear circuit equations the methods most used are iterative methods particularly NR into DC analysis. The following essential advantages of the NR method have to be emphasized:

- a good (quadratic) local convergence in the solution vicinity,

- reliability when implemented with a step-limiting mechanism in cases of circuits including typical, monotonic and regular nonlinear characteristics when iterations start from points not too far from the solution,

- simplicity in practical implementation [8].

These methods are robust and quadratically convergent when a good starting point, enough close to a solution is supplied. Nonetheless the NR methods sometimes fail because of the difficulty of finding a point sufficiently close to an unknown solution.

Specifically by users of **SPICE** by the **.nodeset** statement is sufficiently close to the actual solution. Unfortunately the user either by not know the solution or may not propose an approximate start point to the solution [2, 3]. Difficulties with convergence may be especially hard if the simulator utilizes more complex nonlinear models, which present a better approximation of real devices [9]. The most important nonlinear circuit is the transistor, which can be modeled with high accuracy. In the above, the methods of analysis have assumed the existence of a unique solution, the most circuit simulator are based on previous assumption.

1.2.4 Mathematical representation of electronic devices

One of the first requirements for performing nonlinear circuit analysis is to define unambiguously an adequate set of modeling circuit primitives witch allows the transistor circuit to be analyzed by computer programs.

A nonlinear circuit to be modeled for computer-aided analysis can be described in different ways:

1. As analytical expression, is given as a unique mathematical function that depends on one or more circuit variables.

2. As a table of numerical data, relating circuit variable for discrete values of the independent variables.

3. As a piecewise-linear model witch approximates the actual element characteristics by linear segment in different operating regions [9, 12].

It is well known that nonlinear circuits consisting of an arbitrary number of linear resistor and diodes possess at most one DC operating point. Several fundamental results relate the topology of a transistor circuit to the number of possible DC operating point. Many transistor circuits are known to possess a unique DC operating point due to their topology alone. Any circuit that does not posses a feedback structure possesses a unique operating point. A circuit that contains more than two transistor may possess numerous operating points. Most methods of analysis nonlinear circuits are based on the assumption that there exist a unique solution of the circuit problem. Some switching circuits, such as flip-flops and schmitt triggers, are know to possess two stable states. Then the circuit analysis problem must be therefore allow more than one solution.

Parameter embedding methods, also known as continuation methods or homotopy methods are robust and accurate numerical techniques employed to solve nonlinear algebraic equations. They are used to find multiple solutions of equations that possess multiple solutions. The main disadvantage of homotopy methods is their implementation complexity and computational intensity.

Computational difficulties to establish the DC operating point of transistor circuits are exacerbated by the exponential nature of the diode-type nonlinearities that model semiconductor devices. Since the traditional methods for solving nonlinear equations describing transistor circuits often exhibited convergence difficulties application of more sophisticated mathematical techniques and tools such as parameter embedding methods, continuation, and homotopy methods.

1.3 Computer algorithms

When a problem is solved through the use of a digital computer, an algorithm must be developed, due to the complexity of the models that describe the behavior of modern electronic circuits and heuristic approximation techniques that use as intermediate evaluation steps to arrive at an acceptable solution is needed. The use of a digital computer is subject to certain physical limitations, computer can perform a limited number of operations arithmetic operations (addition, subtraction, multiplication and division on finite, rational numbers). Hence, other mathematical operations, such as differentiation and integration, must be reduced to above elementary operations. Numerical analysis offer numerical methods suitable for solving real-world problems by using the computer where the main advantage a numerical approach is that, if a solution exist, we can find a numerical procedure for the solution; although in a numerical approximation numerical errors may occur, these errors modified the accuracy of the solution and many of the presented numerical problems are closely related to nonlinear problems. Iterative methods form a potent class of numerical methods for solving physical problems, particularity when nonlinearities are involved. An iterative method is based on the precess generates a sequence of new approximations which hopefully converge to the correct solution, at each step of approximation is called iteration, when the iterations lead to closeness to the solution we say that the method converges for iterative methods are used to solve non-linear problems that have the form (1.1)

Iterative methods is a class of numerical methods for solving nonlinear problems. An iterative method generates a sequence successive approximations, using an initial point of the solution is necessary to continue a sequence of new approximations which converge to a correct solution where approximation step is called an iteration and when the successive iterations are close to the correct solution, then the method converges [9].

1.4 Continuation or homotopy methods

The disadvantage of Newton's method for solving sets of nonlinear equations is that its convergence is only local and may require a very good initial guess of the solution. Homotopy methods have been proposed as an alternative for obtaining global convergence in hard problems where Newton's method fails. Homotopy method is based to replace the nonlinear problem by an easy problem which is trivial to solve. Next the easy problem is deformed continuously into the original one, and during this deformation the path followed by the solution is traced numerically. To finding a circuits DC operating points a

method used is the method of homotopy or continuation. This method entails embedding a continuation parameter λ into a set of nonlinear equations $H(x, \lambda)$. Numerical continuation method are techniques for numerically approximating a solution curve, which is implicitly defined by an undetermined system of equations. In the literature of numerical analysis, the terms numerical continuation and path following are used interchangeably. The solution curve is characterized by initial point, the initial value problems to trace the solution curve, a numerical method is used. However, in general this is not an efficient approach, since it ignores the contractive properties which the curve has in view of the fact that it satisfies the homotopy equation. Instead a typical path following method consist of a succession of two different steps:

- Predictor step: An approximate step along the curve, usually in the general direction of the tangent of the curve. The initial value problem provides motivation for generating predictor steps in the spirit of the technology of numerical solution of initial value problems.

- Corrector steps: One or more iterative steps which ai to bring the predicted point back to the curve by iterative procedure (typically of Newton of gradient type) for solving Homotopy curve [1].

The classical embedding methods assume that the solution path is parametrized with respect to an explicit parameter which is identified with the last variable in Homotopy curve. The equation is presented in the form

$$H(x, \lambda) = 0 \tag{1.2}$$

In general, homotopies used in continuation methods can be divided into two main categories, Natural Parameter homotopies and Artificial Parameter homotopies. There are different homotopy functions, basically all need a starting point, the homotopy method introduce an extra variable and modify original system of equation and to obtain a new system of equations. Geometrically the solutions of the augmented system can be interpreted as show Figure 1.2; where solutions S_1, S_2 and S_3 are located at intersections of the curve with the hyperplane $\lambda = 1$.

In the continuation method, an extra variable x_{n+1} in addition to system of n nonlinear equations in n variables $x = (x_1, x_2, \ldots, x_n)$ and modify a number of equations as:
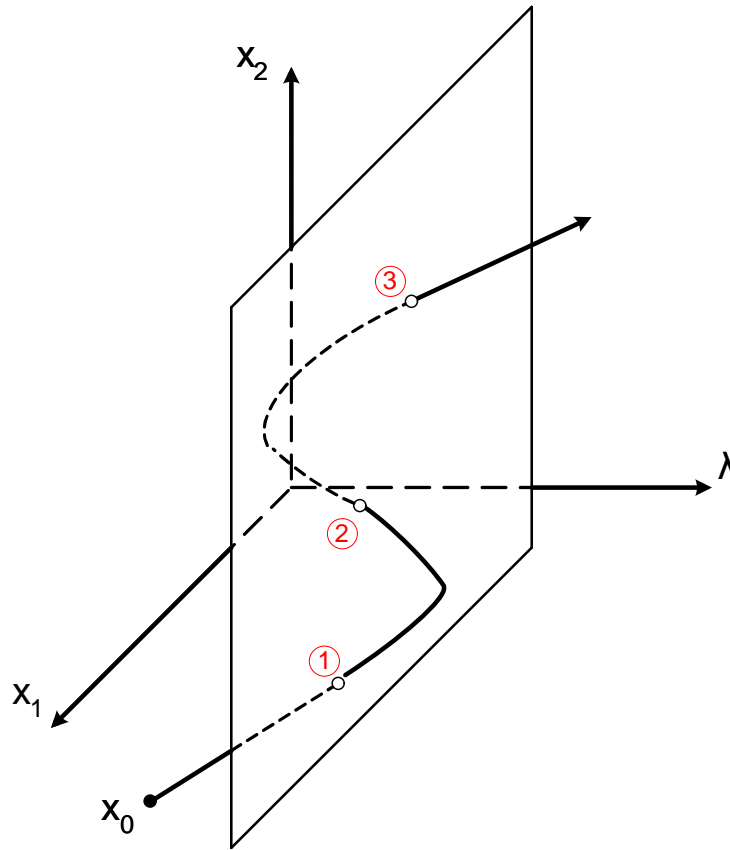
$$F(x, \lambda) = 0, \tag{1.3}$$

9

Figure 1.2: Path tracking of a homotopy curve

is obtained a new system of n equations with $n+1$ variables, when $\lambda = 0$ the trivial solution is found and when $\lambda = 1$ the continuation method allows all solutions lying on the solution curve to be found. The solution curves associated with many practical problems, turn so rapidly that existing algorithms either fail to converge, or become exceedingly slow. The homotopic curve can be described by a parametric equation, where the curve to be the arc-length starting from x_0 initial point. The equation (1.4) is added to the system of equations for solving method based on NR can be used

$$(ds)^2 = (dx_1)^2 + (dx_2)^2 + \ldots + (dx_n)^2 \tag{1.4}$$

1.5 Motivation

In homotopy methods the computational efficiency depends on the homotopy function as well as the curve-tracing algorithm and the initial solution algorithm [4]. For tracing the curve solution the algorithm must be simple to understand and easy to program,

the spherical algorithm the requirements and enables adjustments. In this algorithm the sphere enclosed a part of the solution curve and always intersects the solution curve at least two times then the corrector equation always has at lest two solutions. Spherical algorithm is based on the predictor-corrector steps, where path tracking must be robust, in order to ensure the calibration curve for each iteration. Nonetheless numerical problems can occur such as reversion problem which traces curve to a reverse direction, causing repeated solutions also canceling the characteristics of the formulation homotopic path as well as the tracking error, CPU time and a large number of iterations. Above reasons require research to improve a method of tracing that has characteristics of a simple implementation such as the spherical method.

1.6   Objective of the thesis

This research work is focused on to modify the algorithm for spherical predictor corrector steps in order to avoid the reversion problem. The idea is based on the calculation of the angle formed between the normal vector to the surface of the sphere and the variables forming the plane. Additionally, it is proposed that the step size is variable using the radius of curvature and the hyperbolic tangent function, that allows the curve to go more slowly in the nonlinear part of the curve and runs more quickly when the curve exhibits less non-linearities.

This work is organized as follow. In Chapter 2 are presented different formulations of homotopy that exist in the literature as well as the traditional method of hyperspheres. To the following Chapter modification occurs that make the path of the curve is efficient with respect to the path of the curve and improve iteration number and CPU time; the above using modifications to the problem of reversion and reducing the number of iterations and the computational time. Further Chapter 4 contains different case studies that show the usability of the improve spherical method; the proposed method is applied to solve problems with 2,3,4,14 and 18 variables to ensure that the modification to the hypersphere method is crrect, and also to show the improvement over the traditional hypersphere method. Finally, in Chapter 5 the conclusions and future work are presented.

1.7 Summary

Find the operating point for a circuit is the first step in the analysis of DC for Non-linear circuits. Nonetheless the the task of solve Non-linear equations is still not solved, so the objective of the doctoral thesis is the proposed Improved spherical continuation algorithm to trace the homotopy path-tracking.

Bibliography

[1] ALLGOWER, E. L., AND GEORG, K. Continuation and path following. *Acta Numerica 2* (1 1993), 1–64.

[2] GREEN, M. An efficient continuation method for use in globally convergent dc circuit simulation. In *Signals, Systems, and Electronics, 1995. ISSSE '95, Proceedings., 1995 URSI International Symposium on* (1995), pp. 497–500.

[3] GREEN, M., AND MELVILLE, R. Sufficient conditions for finding multiple operating points of dc circuits using continuation methods. In *Circuits and Systems, 1995. ISCAS '95., 1995 IEEE International Symposium on* (Apr 1995), vol. 1, pp. 117–120 vol.1.

[4] INOUE, Y., IMAI, Y., ANDO, M., AND YAMAMURA, K. A nonlinear homotopy method for solving transistor circuits. In *Communications, Circuits and Systems, 2004. ICCCAS 2004. 2004 International Conference on* (2004), vol. 2, pp. 1354–137B Vol.2.

[5] KUH, E., AND HAJJ, I. Nonlinear circuit theory: Resistive networks. *Proceedings of the IEEE 59*, 3 (March 1971), 340–355.

[6] LAGARIAS, J. C., AND TRAJKOVIC, L. Bounds for the number of dc operating points of transistor circuits. *Circuits and Systems I: Fundamental Theory and Applications, IEEE Transactions on 46*, 10 (Oct 1999), 1216–1221.

[7] MELVILLE, R. C., TRAJKOVIC, L., FANG, S. C., AND WATSON, L. T. Artificial parameter homotopy methods for the dc operating point problem. *IEEE Trans. on CAD of Integrated Circuits and Systems 12*, 6 (1993), 861–877.

[8] OGRODZKI., J. *Circuit simulation: methods and algorithms.* CRC Press, 1994.

[9] SCHWARZ, A. *Computer-aided Design of Microelectronic Circuits and Systems: Digital-circuit aspects and state of the art.*

[10] TRAJKOVIC, L., MELVILLE, R., AND FANG, S.-C. Finding dc operating points of transistor circuits using homotopy methods. In *Circuits and Systems, 1991., IEEE International Sympoisum on* (Jun 1991), pp. 758–761 vol.2.

[11] TRAJKOVIC, L., MELVILLE, R., AND FANG, S.-C. Improving dc convergence in a circuit simulator using a homotopy method. In *Custom Integrated Circuits Conference, 1991., Proceedings of the IEEE 1991* (May 1991), pp. 8.1/1–8.1/4.

[12] YAMAMURA, K., AND OHSHIMAR, T. Finding all solutions of piecewise-linear resistive circuits using linear programming. *Circuits and Systems I: Fundamental Theory and Applications, IEEE Transactions on 45*, 4 (1998), 434–445.

Chapter 2

Homotopy methods and path tracking

Computation of one or more DC operating point of a nonlinear circuit, such as a very large scale integration (VLSI) circuit, is one of the most important and difficult task in an electrical circuit simulator. Most circuit simulators find DC operating points using the NR method, however because of convergence issues the Homotopic continuation methods (HCM) are used an alternative. Homotopic continuation methods (HCM) are proposed as alternative

2.1 Formulation a system of equations

Conventionally, one analyzes a circuit to find its node voltages using Kirchhoffs Current Law (KCL). A node voltage is calculated with respect to a common reference point. Sometimes a branch current is also required; a branch current is the current flowing between two nodes in the circuit. Before continuing, it is important to define KCL and KVL. KCL states, "The sum of currents flowing into and out of a node is zero and KVL states, The sum of branch voltages around a closed loop in any circuit is zero". We formulate equations to represent each branch current and apply KCL to sum the currents at each node. Thus, is obtain simultaneous linear equations which must be solved, as a matrixvector equation, to find the node voltages. The matrix is often called the nodal admittance matrix, which contains the transconductances (partial derivatives of each devices characteristic equations with respect to the circuit variables, i.e., the Jacobian) of nonlinear devices as well as the conductances of linear devices e.g., resistors. The solution vector contains node voltages and possibly branch currents, and the right-hand side (RHS) vector contains the circuit excitations in the form of current sources.

The admittance matrix is normally constructed using element stamps, which are briefly discussed below. Later we will see how evolutionary methods have advantages over this traditional technique. Element stamps are small component-specific tables containing matrix and excitation data. The table indicates the component values to insert in the nodal admittance matrix and in the RHS vector. For example, consider a linear conductance G_{lj} between two nodes l and j, with nodes voltages v_l and v_j. It is similarly possible to define stamps for nonlinear elements, and such stamps form an efficient method for updating the admittance matrix and RHS vector.

2.2 A basic Newton-Raphson method

The Newton procedure considered above will be illustrated graphically as shown in Figure 2.1. In the SPICE-like simulators, the Newton Raphson method is commonly used for solving non-linear equations. This method is based on the following iterative equation:

$$x^{j+1} = x^j - [J(x^j)]^{-1} f(x^j) \tag{2.1}$$

where x^{j+1} is the vector of unknowns variables for current iteration, while x^j are values of the unknown variables in the previous iteration [15]. The following essential advantages
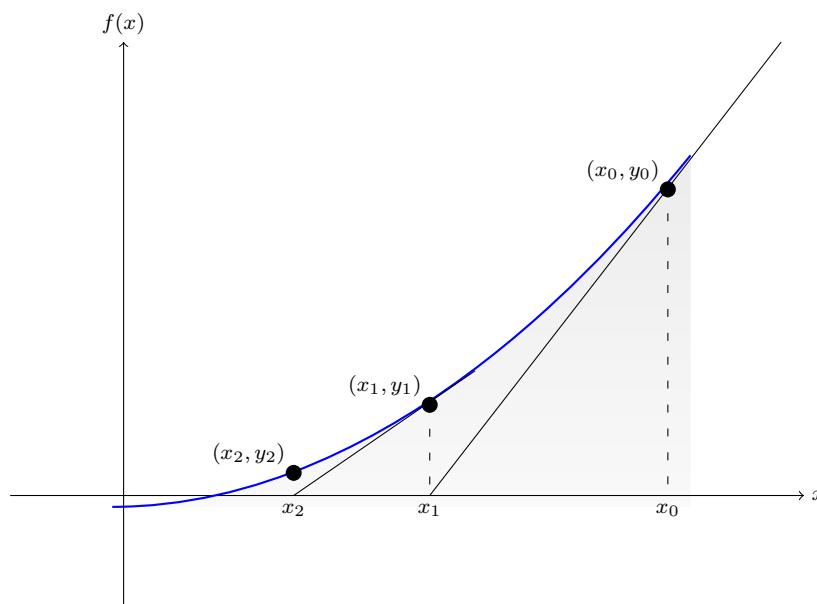

Figure 2.1: Geometric interpretation of the Newton Raphson method

of the NR method have to be emphasized:

- a good local convergence in the solution vicinity, it is if the initial guess is sufficiently close to a correct solution then the Newton-Raphson algorithm will always converge to solution,

- reliability when implemented with a step-limiting mechanism in cases of circuit including typical, monotonic and regular non-linear characteristics when iterations star from points not too far from the solution in a robust implementation of the method,

- simplicity in practical implementation, however the algorithm calculates Jacobian matrix and for the n-dimensional version of the NR the computational work is increases.

Nonetheless although promising advantages occur successfully some weakness must be exposed as disadvantages of the method:

- need for explicit formulae for derivatives,

- weak convergence when starting from a bad approximation to a solution,

- weak convergence in the cases of nonlinearities of rapidly changing slopes, negative slopes, hysteretic ranges, discontinuities of derivatives [17].

2.2.1 Dependent techniques of Newton Raphson method

To help DC convergence experienced designers of analog circuits use several techniques that still depend of NR methods for solving non-linear circuit equations [20].

1. Source-*stepping* algorithm a ramping function is used for the sources and the circuit simulator provides a series of operating points until the response to the original set of driving voltages is obtained.

2. Temperature-*sweeping* procedure, the temperature is swept starting from a accurate value and a DC operating point of the circuit is found. Then the temperature is increased and the new operating point is calculated. The process is repeated until de DC operating point at the desired temperature is found.

3. G_{min}- *stepping* this process consists of small conductances placed at each node of the circuit and ground. The initial value of the conductance is chosen as large as possible to ensure the convergence of NR method. The added conductances contribute to the diagonal element of the circuits jacobian matrix and can force it to become row or column sum dominant. When operating point of the circuit is found, it is used to set

initial node voltage for the next step. The auxiliary conductances are decreased until a default minimum value is achieved. The constant G_{min} stops zeros from occurring on the diagonal, which in turn prevents the matrix from becoming singular. This procedure is equivalent to a large resistance (small conductance) being connected between every node in the circuit and ground.

Some method has been proposed a damping algorithm to aid the convergence of NR; the damping factor α is initialized to a small number where the increments of α should be quite large for the first few iterations but should then decrease gradually. Hence, this technique stops sudden large changes that occur between successive approximate solutions, particularly over the first few iterations. All these techniques rely on the Newton-Raphson method or its variant for solving non-linear circuit equations. They use the idea of the continuation method where a parameter is varied over a range of values until the desired operating point is found. These techniques do not all work for every problem. Hence, commercial simulators employ a number of these techniques and will switch as necessary between them if convergence problems arise. The homotopy approach involves formed a simplified version of the circuit whose operating point is needed, finding an operating point of this easier circuit, then "sweeping" some quantity to generate a trajectory of solutions. The terminus of the sweep is the operating point of the original circuit. Once the operating points problem has been formulated as the solution of a system of non-linear equations, the idea of sweeping can be made mathematically precise, and numerical algorithms have been developed to follow the solution trajectories [14]. The continuation methods also called homotopy methods are robust and accurate numerical techniques employed to solve non-linear algebraic equations. They are used to find multiple solutions [16, 13, 25, 27, 26, 14, 8, 9, 21]. Basically homotopy function is formulated from the equations describing the behavior of the system and an additional parameter, obtaining increased the system equations. This system of equations has two solutions when trivial or known lambda equal to zero and the other the original solution of the system. There exist many methods for constructing an augmented equation with the preceding properties. Since the augmented systems has one more unknown than there are equations, there exist in general a continuum of points, geometrically the solutions of the augmented system can be interpreted as collection of space curves [7].

2.3 Homotopy method

The continuation methods are used to find all zeros of a non-linear functions

$$F(x) = 0 \tag{2.2}$$

The homotopy or continuation method, in general this method involve embedding the continuation parameter λ into a set of non-linear equation $H(x, \lambda)$. The continuation approach to finding zeros using a function

$$H(x, \lambda) \tag{2.3}$$

where $H : R^{n+1} \longrightarrow R^n$ and $x \in R^n$ and $\lambda \in [0, 1]$, such that $H(x, 0) \equiv F(x)$ while $H(x, 1) \equiv G(x)$ is a function with known zeros, the zero set

$$\Gamma(H) = (x, \lambda) : H(x, \lambda) = 0 \tag{2.4}$$

is a union of curves and these curves can be individually traced from the known solution set

$$\Gamma_1(H) = x : H(x, 1) = 0 \tag{2.5}$$

to find all solutions Γ_0 of $F(x)$. The function $H(x, \lambda)$ is called a homotopy, and a homotopy path is a path $(x(t), \lambda(t))$ for $t \in [0, 1]$ on which $H(x, \lambda) = 0$. One method to find all the zeros is to choose a homotopy $H(x, \lambda)$ such that each zero of $F(x)$ is on a separate connected component of the zero set of $H(x, \lambda)$, and separate homotopy paths are followed to fiend each zero of $F(x)$. Using homotopy the zeros of $F(x)$ can be located by tracing a curve completely [12]. The solution is found by following a numeric integration procedure, where λ is used as the integration parameter that varies from an initial value λ_0 to a value $\lambda*$ where a solution $x*$, to the original system is found.

2.4 Homotopic formulations

Several homotopy function can be constructed from nodal analysis formulated from Kirchhoff laws.

2.4.1 The Fixed-point homotopy

The formulation of fixed point homotpy is expressed as

$$H(x, \lambda) = (1 - \lambda)G(x - a) + \lambda F(x) \tag{2.6}$$

where, λ is a continuation parameter, a is a random vector and the new parameter $G \in R^n \times R^n$ is embedded. This homotopy has a circuital interpretation in where $F(x)$ are nodal equation describing a transistor circuit, equation (2.6) represents an augmented circuit. A branch consisting of a conductance $\frac{\lambda}{1-\lambda}G$ connected in series to a grounded voltage source a_k is connected to each node for every nodal equation. At $\lambda = 0$, the starting point of the homotopy path, the adder branches include and voltage source, forcing the nodal voltages to be equal to the elements a_k. Then λ increases and added conductances are present in the circuit. When $\lambda = 1$ the added branches are disconnected from the circuit and the augmented circuit becomes identical to the original circuit.

2.4.2 The variable gain homotopy

The variable stimulus homotopy is based on the equation

$$H(x, \lambda) = (1 - \lambda)G(x - a) + F(x, \lambda\alpha) \tag{2.7}$$

where α is a vector consisting of transistor forward and reverse current gain. These current gains are multiplied by λ. Setting $\lambda = 0$ forces all transistor current gains to zero at the beginning at the continuation process. This solution point correspond to the dc operating point of a circuit consisting of resistors and diodes only. Such a circuit always possesses a unique dc operating point, and its dc equations can be solved easily. For example, an efficient way to find the starting point of the homotopy path is to employ the variable-stimulus homotopy to solve the non-linear circuit consisting of resistors and diodes only. Then the variable-gain homotopy is used to find the dc operating points of the original circuit, through small and carefully selected changes of λ, the circuit is slowly deformed and the coupling of the transistors p-n junctions is introduced. For each instance of λ, the circuit equations are solved is found. The original circuit and its dc operating point is obtained when $\lambda = 1$, this combination of variable-stimulus and variable-gain homotopies has been named the hybrid homotopy [19].

2.4.3 The Newton homotopy

The Newton homotopy has similar properties to the Newton Raphson method and it is formulation:

$$H(x, \lambda) = F(x) - (1 - \lambda)F(x_0), \tag{2.8}$$

where $F(x)$ are node equations for the circuit and $F(x_0)$ are node equations for the circuit evaluated at the initial guess. The Newton homotopy has similar properties to the Newton Raphson method, meaning that every homotopy path in regular domain crosses at the same solution point [21].

2.4.4 The Double Bounded Homotopy

The Double Bounded Homotopy (DBH) was proposed by [22] as a homotopy that exhibits closed paths with the following formulation.

$$H(f(x), \lambda) = C(\lambda - a)(\lambda - b) + \exp^{(\lambda-a)(\lambda-b)} \ln(DF^2(x) + 1) \tag{2.9}$$

where $F(x)$ is the original equation system, λ is the continuation parameter, C and D are arbitrary positive constants. This homotopy formulation contain two solution line a and b; when both lines are applied to the homotopy, as a result the trajectory is forced to cross the double solution lines. The Figure shows a closed path with symmetrical branches and two solution lines.

2.5 Convergence of homotopy methods

Choosing a good initial point for the homotopy method is fundamental to guarantee convergence of the homotopy methods. In circuit terms a good initial point may be a solution of a linear circuit. The election of the initial point has influence on the length of the path and the number of iterations to trace the solution curve. Tracing solution curves is a basic problem that is widely encountered in science and engineering. For example, to obtain the driving-point characteristics and transfer characteristics of non-linear resistive circuits, it is necessary to solve and equation of the form (2.10), where x_{n-1} denotes either the driving
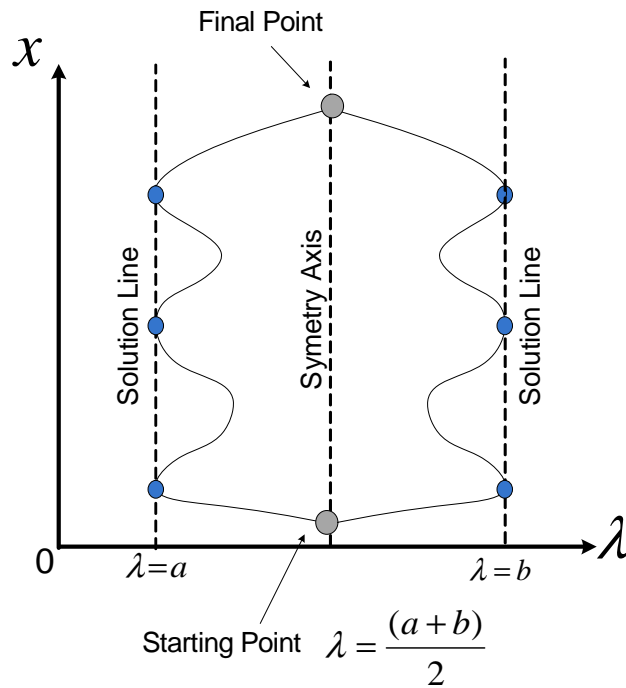
Figure 2.2: DBH Homotopy

point voltage or current [10].

$$
\begin{aligned}
F_1(x_1, x_2, \cdots, x_n, x_{n+1}) &= 0, \\
F_2(x_1, x_2, \cdots, x_n, x_{n+1}) &= 0, \\
F_3(x_1, x_2, \cdots, x_n, x_{n+1}) &= 0, \\
&\vdots \\
F_n(x_1, x_2, \cdots, x_n, x_{n+1}) &= 0,
\end{aligned}
\tag{2.10}
$$

Then applying the homotopy method to (2.10) the following system of equations is obtained by

$$
\begin{aligned}
H_1(F_1(x), \lambda) &= 0, \\
H_2(F_2(x), \lambda) &= 0, \\
&\vdots \\
H_n(F_n(x), \lambda) &= 0,
\end{aligned}
\tag{2.11}
$$

where (2.11) contain n number of equations and $n + 1$ are number of unknown variables. This causes the system can not be solved using conventional algebraic methods. The

24

predictor-corrector algorithm is based on two steps

- Predictor steps. An approximate step along the curve, usually in the general direction of the tangent of the curve. The initial value problem provides motivation for generating predictor steps in the spirit of the technology of numerical solution of initial value problems.

- Corrector steps. One or more iterative steps which aim to bring the predicted point back to the curve by iterative procedure (typically of Newton or gradient type) for solving (2.11).

2.6   Path tracking homotopy

Homotopy methods require suitable path tracking to accurately trace the homotopy curves; otherwise the homotopy simulation can miss solutions or not find any solution at all. Nonetheless the implementation of standard path-tracking techniques is a difficult task [4, 5, 2, 3, 24, 1]. The path tracking based on the predictor-corrector algorithm, its can approximate the curve using tangents lines while in the corrector step an extra equation is needed in the system (2.11). Where the equation is formed by the differentiation of $H(x,t)$ with respect to arclength along the solution path. The Figure 2.3 shows the predictor point for (x^j, λ_j) calculated from the equation,

$$(\bar{x}^{j+1,\bar{\lambda}_{j+1}}) = (x^j, \lambda_j) + h * t \tag{2.12}$$

where h is step length and t is a normalized tangent vector to the homotopy trajectory. After finding the predictor point is necessary to return to the homotopy trajectory by using a corrector step (x^{j+1}, λ_{j+1}). To path tracking the homotopy curve a starting point is
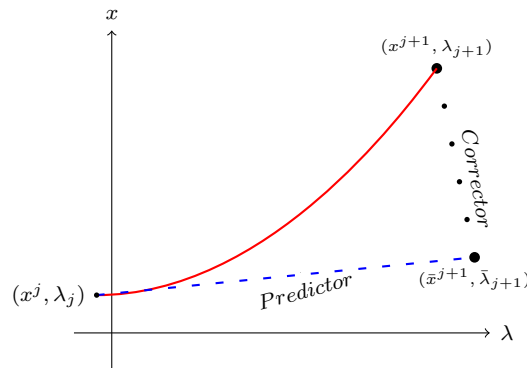

Figure 2.3: Predictor-corrector scheme

proposed X_0, then the predictor-corrector scheme is applied to find a solution implementing zero strategy, now repeat the process taking as starting point the solution found in previous iteration. The dc driving-point and transfer characteristics of nonlinear circuits are the multivalued curves that arise from the nature of the circuit. These curves cannot be analyzed by general-purpose circuit simulators. One known method for analyzing these kinds of characteristic curves is the backward differentiation formula (BDF) curve-tracing algorithm proposed by [6]. In this method, the circuit equations $f(x) = 0$, where the input voltage is assumed to be a variable, are analyzed by the predictor-corrector algorithm where the arc-length of the solution curve in $n + 1$ dimensional space is the parameter. However, it is not clear that this method is practical for large-scale circuits. The method of hyperspheres uses the predictor-corrector scheme for implementation, where the spheres are used for plotting the curve of homotopy [11, 23].

2.7 Hyperspherical algorithm for tracing curves

Mentioned previously algorithm for tracing solution curves that uses hyperspheres. Each sphere encloses the curve and has at least 2 intersections (the corrector equation has at least 2 solutions). Where $(c_1, c_2, \cdots, c_{n+1})$ are points on the solution curve and are also the centers for spheres. The intersections of the sphere and the solution curve are solutions of the following system of $n + 1$ equations:

$$
\begin{aligned}
&F_1(x_1, x_2, \cdots, x_n, x_{n+1}) = 0, \\
&F_2(x_1, x_2, \cdots, x_n, x_{n+1}) = 0, \\
&F_3(x_1, x_2, \cdots, x_n, x_{n+1}) = 0, \\
&\vdots \\
&F_n(x_1, x_2, \cdots, x_n, x_{n+1}) = 0, \\
&(x_1 - c_1)^2 + (x_2 - c_2)^2 + \cdots + (x_{n+1} - c_{n+1})^2 = r^2
\end{aligned}
\tag{2.13}
$$

Curve is traced solving the system of nonlinear equations of the form (2.13). In each iteration the solution that has just been obtained is used that the center of the new sphere. During the corrector step, the radius of the sphere is shortened if Newtons method does not converge. Corrector equation has at least two solutions one of witch is situated in the forward direction and another of witch situated in the backward direction. Using sufficiently small spheres, Newton method will converge to the rightful solution on the corrector step. Where rightful solution is situated in the forward direction (Forward Solution FS) . If the solution is located in the backward direction (Backward Solution BS)
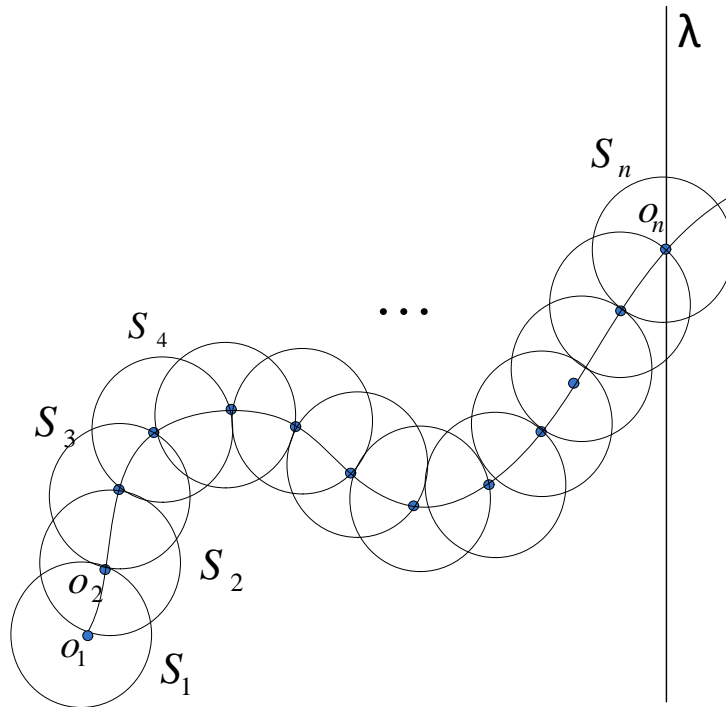
Figure 2.4: Solution curves with spheres

then the reversion of the curve tracing appears and is called "reversion phenomenon" (see 2.5). If the radius of the new sphere is the same as the previous one, the reversion solution can be detected because the reversion solution is the same as the previously obtained solution. Predictor step can be considered as a step of the Euler method for solving the differential equation that describes the homotopy trajectory. Predictor step is based on tangent predictions, secant predictor, interpolation predictor, Taylor polynomial predictor and corrector step is based in NR method.

2.8   Find zero strategy

When the curve is tracked past $\lambda = 1$, the algorithm enters an "end game" [18], phase to compute the solution in the hyperplane $\lambda = 1$. The algorithm start from two points very near to $\lambda = 1$,then the points are interpolated to obtain a closest point to $\lambda = 1$ to have a closest point to the solution. The interpolation can de used Hermite cubic.
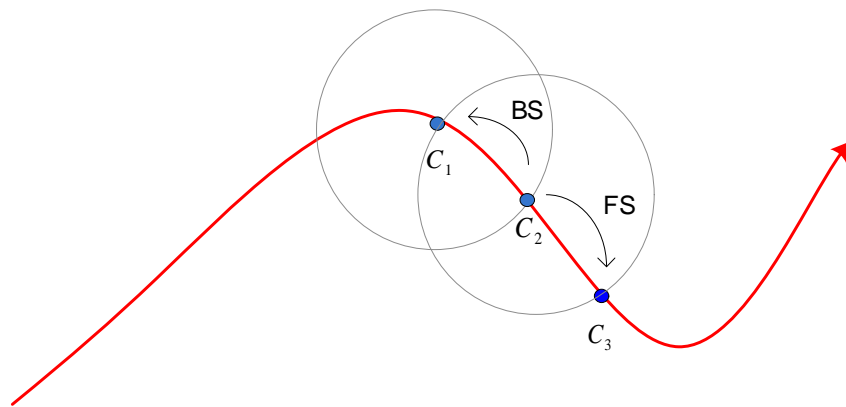
Figure 2.5: Forward Solution (FS) and Backward Solution (BS)

2.9   Summary

The spherical method to trace the solution curve is simple a geometrically clear. Nonetheless reversion phenomenon during the trace of the curve causes problems in tracing, finding solutions previously found and spending computation time. The contributions in this thesis are the proposition and implementation of improvements to the hyperspherical method. The improvements are the methodology to detect the phenomenon of reversion and correct the trajectory and a methodology for varying the radius of the sphere conforming to the shape of the curve homotopic.

Bibliography

[1] ALLGOWER, E., AND K.GEORG. *Numerical path following.* 1994.

[2] ALLGOWER, E. L., AND GEORG, K. Continuation and path following, 1992.

[3] BATES, D. J., HAUENSTEIN, J. D., AND SOMMESE, A. J. Efficient path tracking methods. *Numerical Algorithms 58*, 4 (2011), 451–459.

[4] BATES, D. J., HAUENSTEIN, J. D., SOMMESE, A. J., AND WAMPLER, C. W. Ii, adaptive multiprecision path tracking. *SIAM J. Numer. Anal. 46 (2).*

[5] BATES, D. J., HAUENSTEIN, J. D., SOMMESE, A. J., AND WAMPLER, C. W. Stepsize control for adaptive multiprecision path tracking. *Contemp. Math. 496.*

[6] CHUA, L. O., AND USHIDA, A. A switching-parameter algorithm for finding multiple solutions of nonlinear resistive circuits. *International Journal of Circuit Theory and Applications 4*, 3 (1976), 215–239.

[7] CHUA, L. O., U. A. A switching-parameter algorithm for finding multiple solutions of nonlinear resistive circuits. In *International Journal of Circuit Theory and Applications* (1976), vol. 4, pp. 115–239 vol.4.

[8] DYES, A., CHAN, E., HOFMANN, H., HORIA, W., AND TRAJKOVIC, L. Simple implementations of homotopy algorithms for finding dc solutions of nonlinear circuits. In *ISCAS (6)* (1999), pp. 290–293.

[9] INOUE, Y., IMAI, Y., ANDO, M., AND YAMAMURA, K. A nonlinear homotopy method for solving transistor circuits. In *Communications, Circuits and Systems, 2004. ICCCAS 2004. 2004 International Conference on* (2004), vol. 2, pp. 1354–137B Vol.2.

[10] K., Y. Simple algorithms for tracing solution curves. *Circuits and Systems I: Fundamental Theory and Applications, IEEE Transactions on 40*, 8 (1993), 537–541.

[11] K., Y. Simple algorithms for tracing solution curves. *Circuits and Systems I: Fundamental Theory and Applications, IEEE Transactions on 40*, 8 (1993), 537–541.

[12] LAGARIAS, J. C., REEDS, J. A., WRIGHT, M. H., AND WRIGHT, P. E. Convergence properties of the nelder–mead simplex method in low dimensions. *SIAM J. on Optimization 9*, 1 (May 1998), 112–147.

[13] LEE, J., AND CHIANG, H.-D. Constructive homotopy methods for finding all or multiple dc operating points of nonlinear circuits and systems. *Circuits and Systems I: Fundamental Theory and Applications, IEEE Transactions on 48*, 1 (2001), 35–50.

[14] MELVILLE, R. C., TRAJKOVIC, L., FANG, S. C., AND WATSON, L. T. Artificial parameter homotopy methods for the dc operating point problem. *IEEE Trans. on CAD of Integrated Circuits and Systems 12*, 6 (1993), 861–877.

[15] OGRODZKI., J. *Circuit simulation: methods and algorithms.* CRC Press, 1994.

[16] RHEINBOLDT, W. C., AND BURKARDT, J. V. A locally parameterized continuation process. *ACM Trans. Math. Softw. 9*, 2 (June 1983), 215–235.

[17] SCHWARZ, A. *Computer-aided Design of Microelectronic Circuits and Systems: Digital-circuit aspects and state of the art.*

[18] SOSONKINA, M., WATSON, L. T., AND STEWART, D. E. Note on the end game in homotopy zero curve tracking. *ACM Trans. Math. Softw. 22*, 3 (Sept. 1996), 281–287.

[19] TRAJKOVIC, L. Homotopy methods for computing dc operating points.

[20] TRAJKOVIC, L., MELVILLE, R., AND FANG, S.-C. Finding dc operating points of transistor circuits using homotopy methods. In *Circuits and Systems, 1991., IEEE International Sympoisum on* (Jun 1991), pp. 758–761 vol.2.

[21] USHIDA, A., YAMAGAMI, Y., NISHIO, Y., KINOUCHI, I., AND INOUE, Y. An efficient algorithm for finding multiple dc solutions based on the spice-oriented newton homotopy method. *Computer-Aided Design of Integrated Circuits and Systems, IEEE Transactions on 21*, 3 (2002), 337–348.

[22] VAZQUEZ-LEAL, H., HERNANDEZ-MARTINEZ, L., AND SARMIENTO-REYES., A. Double-bounded homotopy for analysing nonlinear resistive circuits. In *Circuits and Systems, 2005. ISCAS 2005. IEEE International Symposium on* (2005), pp. 3203–3206 Vol. 4.

[23] VAZQUEZ-LEAL., H., SHEISSA., R. C., BERNAL., F. R., L. HERNANDEZ-MARTINEZ., A. S.-R., AND FILOBELLO-NINO-A., U. Powering multiparameter homotopy-based simulation with a fast path-following technique. *ISRN Applied Mathematics 2011*, 1 (2011), 7 pages.

[24] WATSON, L. T., BILLUPS, S. C., AND MORGAN, A. P. Algorithm 652: Hompack: a suite of codes for globally convergent homotopy algorithms. *ACM Trans. Math. Softw. 13*, 3 (1987), 281–310.

[25] WOLF, D., AND SANDERS, S. R. Multiparameter homotopy methods for finding dc operating points of nonlinear circuits. *Circuits and Systems I: Fundamental Theory and Applications, IEEE Transactions on 43*, 10 (1996), 824–838.

[26] WU, T.-M. Solving the nonlinear equations by the newton-homotopy continuation method with adjustable auxiliary homotopy function. *Applied Mathematics and Computation 173*, 1 (2006), 383–388.

[27] YAMAMURA, K., SEKIGUCHI, T., AND INOUE, Y. A fixed-point homotopy method for solving modified nodal equations. *Circuits and Systems I: Fundamental Theory and Applications, IEEE Transactions on 46*, 6 (1999), 654–665.

Chapter 3

Proposed methodology for the variable radius in the spherical method

3.1 Introduction

In this chapter is presented an Improved Spherical Method to trace the path homotopy using hyperspheres. Notwithstanding for implementation to path tracking algorithm used in homotopy continuation system exhibits advantages and disadvantages. The reversion problem is an issue identified as disadvantage for The Spherical Method. Such problem must be attacked to improve the path to find the solutions. Then another significant disadvantage is the the computation time spent to trace the solution curve. The strategy to avoid the problem of reversion is established, besides is proposed a methodology to trace the homotopy path using different values for the radius of the sphere for the spherical method. The above reduce the number of iterations for path homotopy.

3.2 Reversion problem in Spherical algorithm

The Spherical Algorithm is implemented using Predictor-Corrector to path tracking, the proposed scheme is described as follow: The proposed MSA scheme is described as follows:

- Predictor: The predictor point is given using Figure 3.1. Using the point o_1 as the center of the first sphere and o_2 as the center of the second sphere; prediction point is calculated by obtaining k_1. The last point will be used as initial point for the NR method until find the point o_3 used as the center of the next sphere.
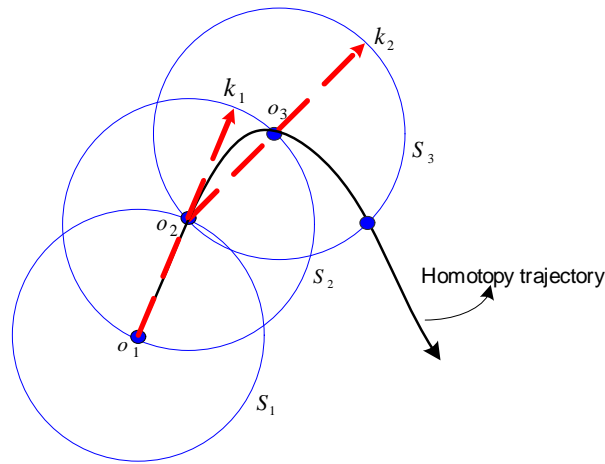
Figure 3.1: Hyperspheres algorithm
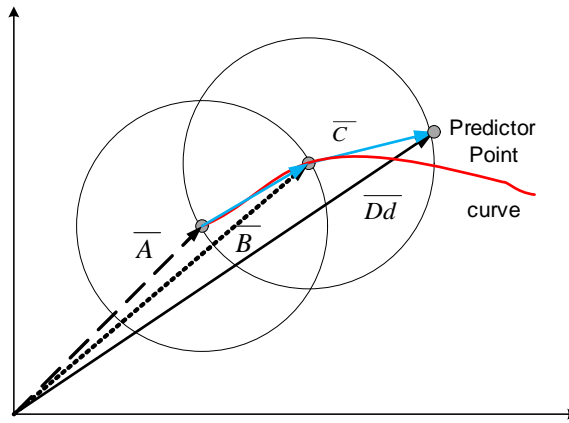
Figure 3.2: Predictor point.

The step predictor is calculated by addition of vectors by Figure 3.2, where A contain previous solution B is a center of the sphere and Dd is the predictor point.

The implementation of standard path-tracking techniques is a difficult task; the Spherical Algorithm has a clear geometrical interpretation, which facilitates its implementation in programming. The Spherical Algorithm use spheres of dimensions $n + 1$, where $n + 1$ is the number of variables in a set of nonlinear equations, such spheres are allocated over the solution curve. The spheres used in the Spherical Algorithm have at least two intersections over the homotopy path. The homotopy formulation contains n equations and $n + 1$ variables, where $x_i(i = 1, ..., n)$ represent the variables of the system and x_{n+1} is the homotopy parameter λ. So here the number of equations is less than the number of variables; to solve this issue is added the sphere equation. The equation that describes

34

the behavior of the sphere with center c and radio r is represented by

$$S(X) = (x_1 - c_1)^2 + (x_2 - c_2)^2 + ... + (x_{n+1} - c_{n+1})^2 - r^2 = 0. \tag{3.1}$$

Then the augmented system using homotopy formulation contains the same number of equations as variables using the (3.1)

$$\begin{aligned}
H_1(F_1(x), \lambda) &= 0, \\
H_2(F_2(x), \lambda) &= 0, \\
&\vdots \\
H_n(F_n(x), \lambda) &= 0, \\
S(x_1, x_2, \cdots, x_n, \lambda) &= 0.
\end{aligned} \tag{3.2}$$

By solving the system of equations (3.2) using the Newton Raphson method the intersection of the sphere with the curve is found, updating the center of following sphere during traced of homotopy path. The spheres $(S_1, S_2, \cdots)$ are located in succession along the path following of the solution curve. Figure 3.3 contain at least two solutions, the solution o_4 is in forward direction and solution o_2 is in reverse direction. Solution in the reverse direction is a comeback for the traced homotopy curve, which caused a problem to programming implementation of the spherical algorithm. The Figure 3.4 shows the
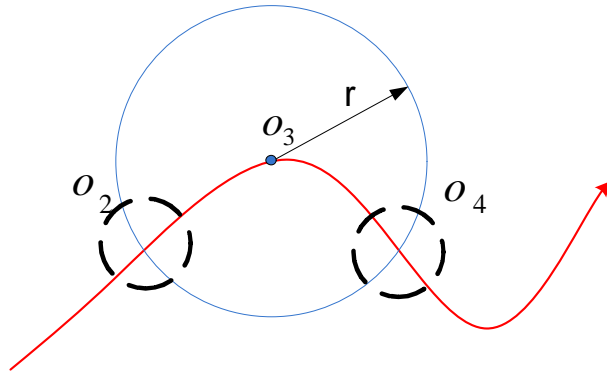
Figure 3.3: Sphere intersecting with the curve at two points

path by using successive spheres. where $(S_1, S_2, \cdots)$ are the needed spheres for tracing the homotopy path to find a solution or solutions and $(O_1, O_2, \cdots)$ are consecutive centers for spheres. The "reversion problem" must be attended to guarantee that the path has been traced completely, the following section shows a solution proposed from detection and correction of the reversion problem. The Figure 3.5 shows a successful case of forward path.
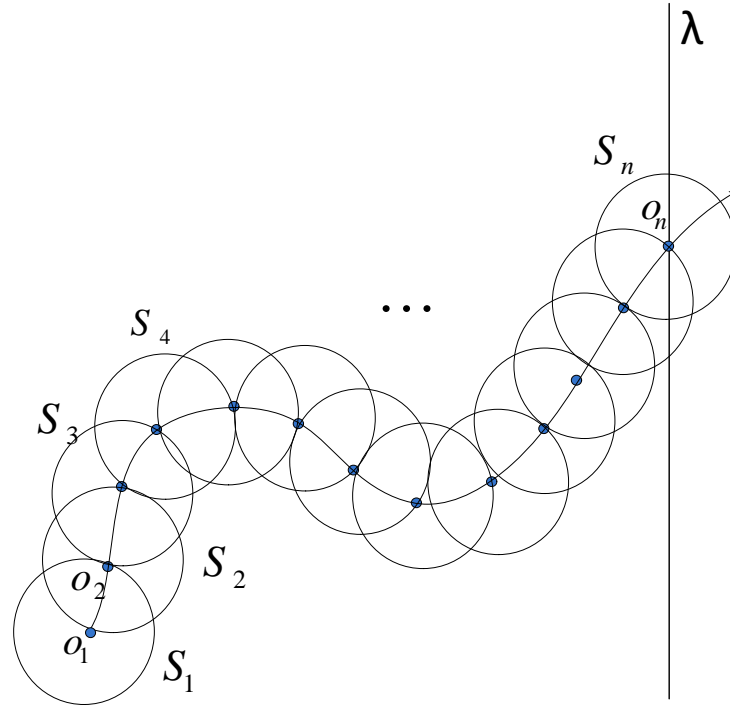
Figure 3.4: Solution curves with spheres

3.3 Strategy to detect the reversion phenomenon

This section describes the proposed strategy to detect the reversion problem to trace path tracking. Mainly the problem of ensure the existence of reversion phenomenon resides in numerical problems. The successive intersections of the sphere with the solution curve are numerically close. Using fixed radius for the spheres could aid help detect the reversion phenomenon. Nonetheless, the backward and forward solutions are really close numerically, making it difficult to differentiate between then. The strategy proposed is based on calculating normal vector angles for the solution (o_2, o_4) of the sphere and is compared the angles instead of numerical solutions [2, 1]. As already stated, determining the differences between the forward direction o_4 and the backward direction o_2 it is not an easy task. Therefore, calculating normal vector angles for both solutions o_2 and o_4 as depicted Figure 3.6 can help solve to discern among solutions. For conducting this proposal begins with the calculation of the gradient of the sphere equation (3.3)

$$\nabla S = S'_{x_1}\hat{x}_1 + S'_{x_2}\hat{x}_2 + \cdots + S'_{\lambda}\hat{\lambda} \tag{3.3}$$

The gradient is a generalization of the usual concept of derivative to the functions of several variables and represents the slope of the tangent of the graph of the function.
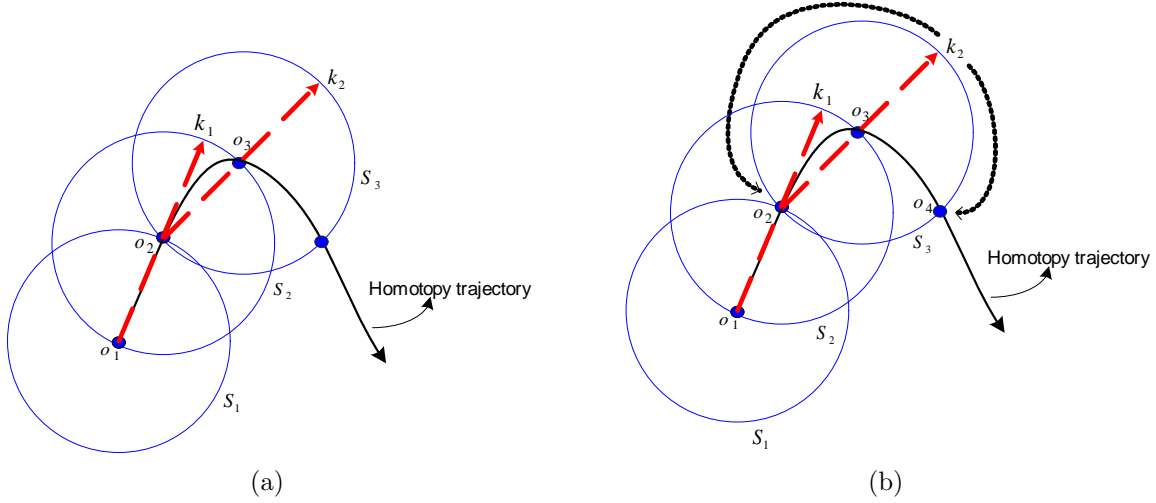
Figure 3.5: (A) Forward path. (B) Path back.

In others words is the derivative of each of the system variables and λ. Then using the gradient of the sphere equation the normal vector $\vec{n}$ is obtained by

$$\vec{n} = \frac{\nabla S}{\|\nabla S\|} \tag{3.4}$$

The notation $\|\nabla S\|$ represents the Euclidean norm of ∇S. Then as mentioned previously its necessary to obtain the angle of $\vec{n}$ calculate by

$$\theta_{x_i} = \cos^{-1} \frac{S'_{x_i}}{\|\nabla S\|}, \qquad i = 1, 2, \ldots, n + 1 \tag{3.5}$$

where θ_{x_i} is the angle with respect to the coordinate axis, i correspond to the number of system variables and $n + 1 = \lambda$. The Figure 3.7 shows the normal vector to the sphere $\vec{n}$ with their respective angles for each axis $(\theta_{x_1}, \theta_{x_2}, \ldots, \theta_{\lambda})$. Now, instead of comparing o_2 and o_4 directly, we use the angles of their normal vectors for an efficient comparison. The calculated angles depends on the number of variables in the system, during programming Spherical Method should be done comparing the angles of the previous solution with the angles of the current solution. In the case where the angles are equal then the reversion is detected, otherwise the path is traced.
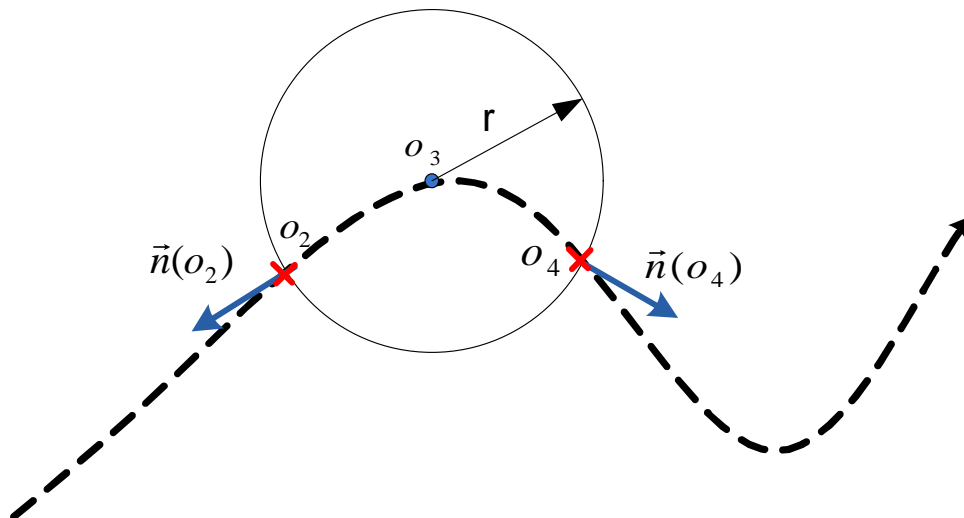
Figure 3.6: Normal vector into the sphere
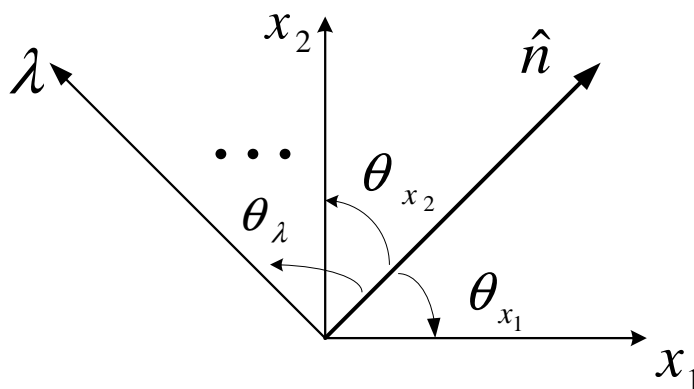
Figure 3.7: Angle to normal vector.

3.4   Strategy to avoid the reversion phenomenon

Now, instead of comparing o_2 and o_4 directly, we use the angles of their normal vectors for an efficient comparison avoiding numerical problems. After detecting the reversion phenomenon, we have to modify the corrector step to get back to the trajectory, by increasing the radius a δr inducing the corrector step to converge to the forward solution. Then, we propose the point $k_2 + \delta r$ that result in k_2' as initial point for NR (see Figure 3.8). This technique creates a perturbation in the corrector step that can induce the convergence to the forward solution. For this work the step size takes values of δr which coincides with the sphere radius r, where δr is an increase in the radius. The proposed to avoid the reversion problem in the spherical method will be tested later using different
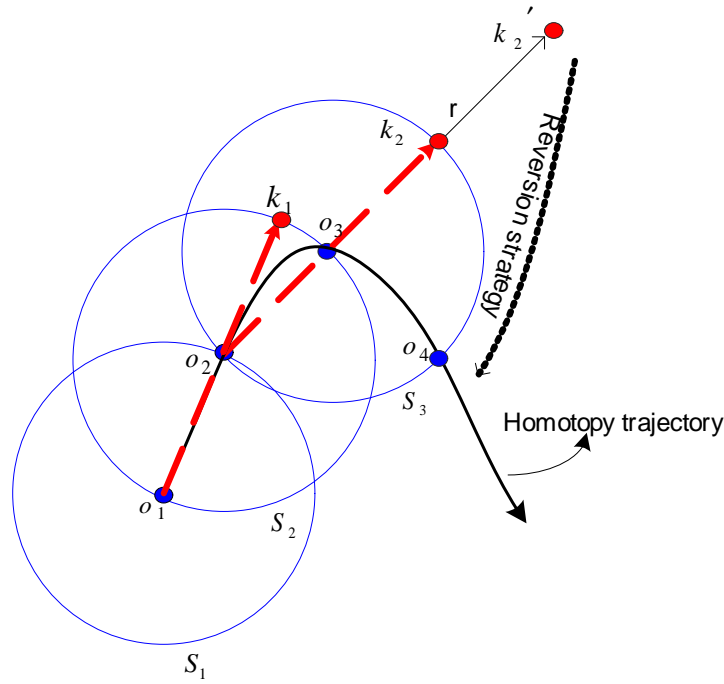
Figure 3.8: Proposed reversion strategy.

study cases. Next is presented a methodology to adapt the radius of the sphere with respect to the shape of the curve achieving fewer iterations for tracing a path homotopy.

3.5 Proposed methodology to accelerate the course of the curve on the spherical algorithm

Homotopy methods are characterized by slowness to the path tracking, such problem caused by the path tracking. This work use the spherical algorithm to trace the homotopy path, in original spherical algorithm consecutive spheres having the same size radius, nonetheless when the curve has a change shape of the path is suitable a decreased radius and when the curve not change shape then radius size must be increased [3]. The proposed strategy in this work consists in changing the radius size for the area based on prior knowledge of the behavior of the curve and its folds; the larger the radius of curvature, the flatter the curvature of the surface. Conversely, the shorter the radius of curvature, the steeper the curvature of the surface, using the calculation of the radius of curvature

provides information on the behavior of the curve obtaining

$$\rho = |\frac{(1 + (x_i')^2)^{\frac{3}{2}}}{x_i''}| \tag{3.6}$$

where x_i', x_i'' represent differentiation of first and second order respectively. Differentiation must be numeric to involve only the points where the sphere intersects the curve and reduce complexity by involving all variables in the system of equations. Numerical differentiation of first order is calculated using:

$$x_i' = \frac{x_i - x_{i-1}}{h} \tag{3.7}$$

x_i and x_{i-1} are a current iteration and a preceding iteration and h is the distance between them (See Figure 3.9).
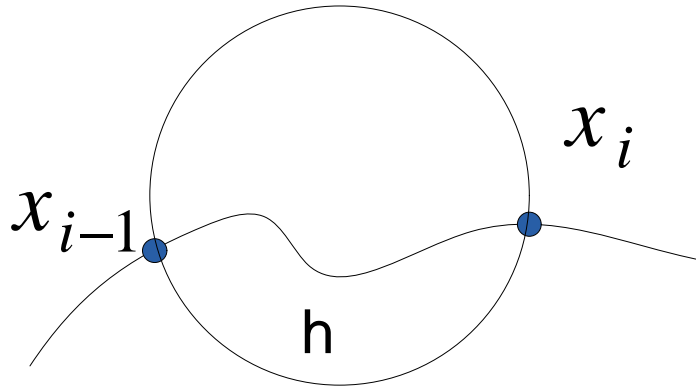
Figure 3.9: Numerical differentiation.

Now the differentiation of second order can be calculated using differentiation of first order as

$$x_i'' = \frac{x_i' - x_{i-1}'}{h} \tag{3.8}$$

where x_i' is first order differentiation evaluated in the current iteration and x_{i-1}' is first order derivative evaluated in the previous iteration.

Using (3.7) and (3.8) to obtain (3.6) for each variable $x_1, x_2, \cdots x_n$ belonging to the system of equations. For calculating the radius of curvature is necessary parametrization of the curve, notwithstanding in this work the radius of curvature for each variable is obtained and then an average of the calculated radius of curvature. Then the average radius of curvature of each system variable is obtained by,

$$\rho_{av} = \frac{\rho_{x_1} + \rho_{x_2} + \cdots + \rho_{x_n}}{n} \tag{3.9}$$
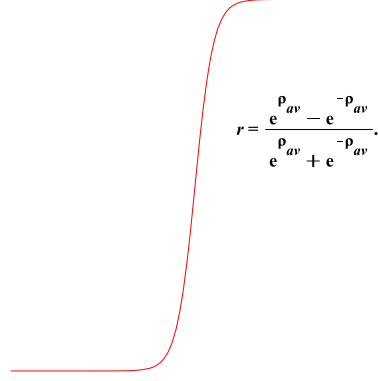
Figure 3.10: Hyperbolic tangent function.

where n is the number of variables. The radius of the sphere is calculated using the radius of curvature (3.9), within an asymptotic function as (3.10).

$$r = f(\rho_{av}) = tanh(\rho_{av}) = \frac{Ae^{\rho_{av}} - Be^{-\rho_{av}}}{Ce^{\rho_{av}} + De^{-\rho_{av}}} \tag{3.10}$$

The asymptotic function has a maximum value and a minimum value that depends on the radius of curvature, the Figure 3.10 shows the function behavior where radio size can take minimum and maximum values depending on the curvature of the path. The value of C and D is 1, l_{up} y l_{low} are values for the radius of the sphere when the radius of curvature tends to hight value or a low value. For the following system of equations with exponential terms will be used to calculate the radius of curvature by,

$$F_1(x_1, x_2) = (4x_1 + 8)e^{-(x_1+2)^2} - x_2^2 + 24x_1 e^{-3x_1^2-(x_2-2)^2} + (12x_1 - 36)e^{-2(x_1-3)^2} - x_2^2 +$$
$$8x_1 e^{-2x_1^2-2(x_2+1)^2} + (6x_1 + 6)e^{-3(x_1+1)^2-3(x_2-2)^2} = 0,$$
$$F_2(x_1, x_2) = (8x_2 + 8)e^{-2x_1^2-2(x_2+1)^2} + 6x_2 e^{-2(x_1-3)^2} - x_2^2 + 4x_2 e^{-(x-1+2)^2} +$$
$$(8x_2 - 16)e^{-3x_1^2-(x_2-2)^2} + (6x_2 - 12)e^{-3(x_1+1)^2-3(x_2-2)^2}. \tag{3.11}$$

Calculating the radius of curvature for the variables x_1, x_2 respectively obtained Thus most of the iterations is maintained a radius 0.1 without exceeding limit and ensuring that when the radius of curvature tends to reduces the the lower limit radius 0.03. The equation for obtain radius of the sphere (3.14).

$$r = f(\rho_{av}) = \frac{0.1e^{\rho_{av}} - (0.03e^{-\rho_{av}})}{e^{\rho_{av}} + e^{-\rho_{av}}} \tag{3.12}$$

To ensure that the function operates properly to estimate the size of the radius of the

Radius of Curvature	radio(r)
0.21	0.07
0.29	0.08
0.81	0.09
1.62	0.1
23.3	0.1

Table 3.1: Numerical solutions to the equation (3.11)

hypersphere the limit is obtained. The radius of curvature is always greater than zero, so when folds are presented in the curve radius of curvature will be close to zero and radius should be closer to the value of 0.03 for the specific case of interval (from 0.03 to 0.1). Calculating the limit of the function when it approaches zero (for values of the radius of curvature very close to zero) the lowest value of the radius is obtained.

$$r = \lim_{\rho_{av} \to 0} \frac{0.1e^{\rho_{av}} - (0.03e^{-\rho_{av}})}{e^{\rho_{av}} + e^{-\rho_{av}}} = 0.03 \tag{3.13}$$

Then the limit of the function is calculated for large values of curvature radius finding the limit equal to 0.1.

$$r = \lim_{\rho_{av} \to \infty} \frac{0.1e^{\rho_{av}} - (0.03e^{-\rho_{av}})}{e^{\rho_{av}} + e^{-\rho_{av}}} = 0.1 \tag{3.14}$$

Calculated limits assure a range of values that are established at the beginning of the path of the curve to avoid falling into numerical problems due to the very small radios or jumps of the curve caused by very large radius sphere. To trace the curve of homotopy (??) applying (3.14) to resizing the radio for spheres obtaining the Figure 3.11. The Figure 3.11 shows the traced of the solution curve where the points separated are spheres traced with a radius 0.1 and the other cases the radius decrease size. Equation (3.9) is a hyperbolic tangent function used to find the appropriate value of the radius of the sphere to the shape of the curve homotopic having as parameter the radius of curvature at each point of the curve. Sphere radio size can increase using the hyperbolic function exponent with a constant K

$$f(\rho_{av}) = tanh(\rho_{av}) = \frac{e^{K\rho_{av}} - 1}{e^{K\rho_{av}} + 1} \tag{3.15}$$

At each iteration the curvature radius is calculated to obtain the radius of the sphere suitable for tracing the trajectory.
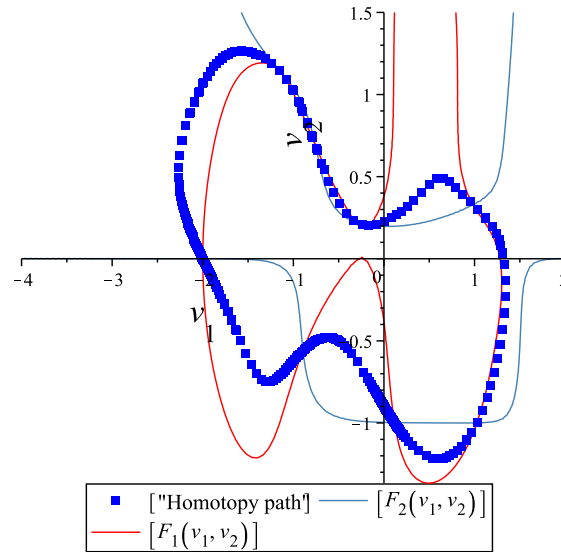
Figure 3.11: Path tracking of studio case.

3.6 Aspects to consider in tracing curve

As mentioned in the previous chapter path tracking starts with $\lambda = 0$ until $\lambda = 1$, where the solutions are encountered. Following the predictor-corrector scheme when it comes to $\lambda = 1$, should be necessary to follow a series of steps until you find the solution.

- Find zero strategy: The finding zero strategy should be established immediately after the path bounces or crossing $\lambda = 1$ depending on the formulation homotopic. The strategy used in this paper is based on a monitoring a change of sign of $\Delta\lambda$ found after the corrector point. Such procedure is realized by multiplying $\Delta\lambda$ of two consecutive predictor step.

$$sign(\Delta\lambda_{j+1}\Delta\lambda_j) \tag{3.16}$$

The sign of $\Delta\lambda$ changes after bouncing from point k_1 to point k_2 as shown in Figure 3.12, where $\Delta\lambda_j = k_2 - k_1$ and $\Delta\lambda_{j+1} = k_2 - k_3$.

Figure ?? shows the solution, which the method applies NR for a more approximate solution.
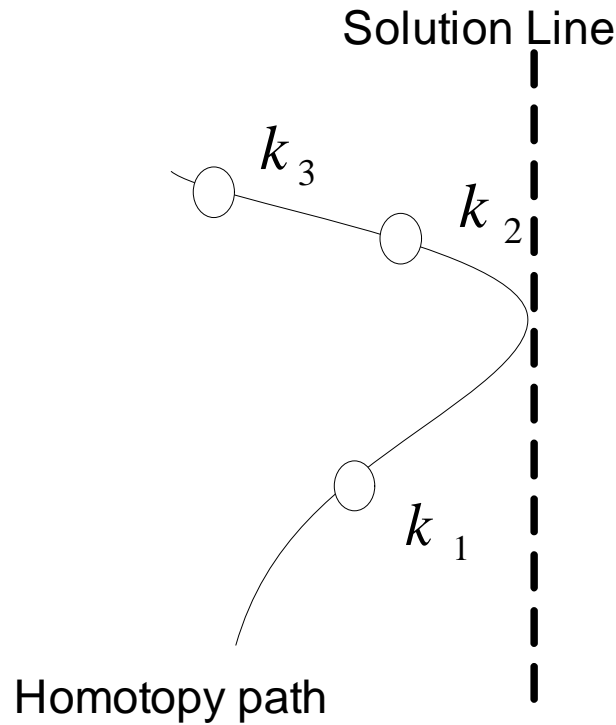
Figure 3.12: Find zero strategy

3.7 Conclusions

The strategies proposed improve tracing the trajectory aforementioned avoiding problems during the implementation of the method of path using spheres. The angles of the normal vector show that the reversion problem can be detected using the proposed methodology. Then after detecting the problem of reversion a correction of the trajectory is applied by increasing the radius of the sphere to increase the likelihood that the solution obtained on the corrector step is the right solution for the sphere.

Solution Line

k_3

k_2

k_1

Homotopy path
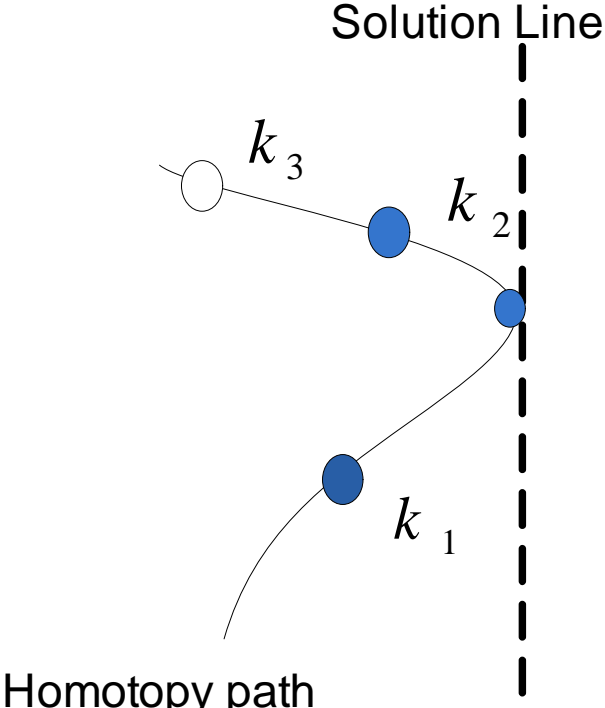
Figure 3.13: Approximate solution

Bibliography

[1] D. Torres-Munoz, L. Hernandez-Martinez, and H. Vazquez-Leal. Improved spherical continuation algorithm by nonlinear circuit. In *Circuits and Systems (MWSCAS), 2013 IEEE 56th International Midwest Symposium on*, pages 61–64, Aug 2013.

[2] D. Torres-Muoz, H. Vazquez-Leal, L. Hernandez-Martinez, and A. Sarmiento-Reyes. Improved spherical continuation algorithm with application to the double-bounded homotopy (dbh). *Computational and Applied Mathematics*, 33(1):147–161, 2014.

[3] Delia Torres-Muoz, Luis Hernandez-Martinez, and Hector Vazquez-Leal. Spherical continuation algorithm with spheres of variable radius to trace homotopy curves. *International Journal of Applied and Computational Mathematics*, pages 1–13, 2015.

Chapter 4

Case studies

In this Chapter, different case studies are presented in order to be solved to improve aspects of computation time, and number of iterations complete path curve homotopy avoiding the problem of reversion. The system of equations to be solved; the strategy to detect the problem of reversion is applied and tested in the first case study to verify its operation.

4.1 Non-linear circuit with two tunnel diodes

The study case shows a circuit with two tunnel diodes, one voltage source and a series resistor as is shown in Figure 4.1 [3, 2].
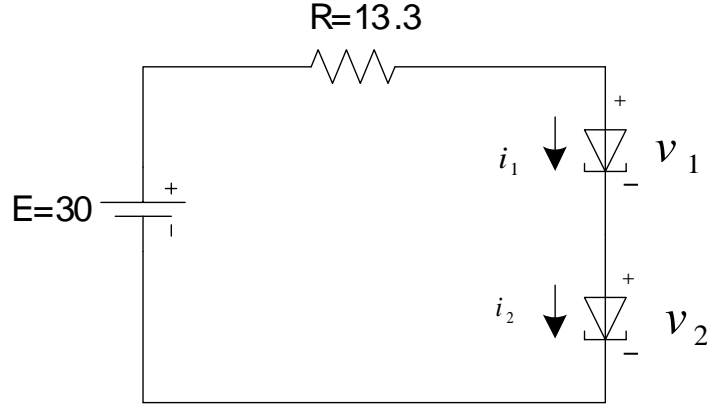
Figure 4.1: Two tunnel diode circuit.

The non-linear expression for each tunnel diode model is shown follow

$$g_1(v_1) = 2.5v_1^3 - 10.5v_1^2 + 11.8v_1,$$
$$g_2(v_2) = 0.43v_2^3 - 2.69v_2^2 + 4.56v_2. \tag{4.1}$$

After applying the Kirchoff laws, we obtain equilibrium equations for the system

$$F_1(v_1, v_2) = E - Rg_1(v_1) - (v_1 + v_2) = 0,$$
$$F_2(v_1, v_2) = g_1(v_1) - g_2(v_2) = 0. \tag{4.2}$$

Applying the DBH homotopy to (4.2), results an augmented system with $n + 1$ equations and $n + 1$ variables

$$H_1(f_1, \lambda) = 40\lambda(\lambda - 1) + \exp(\lambda(\lambda - 1)) \ln(0.01(30 - 33.25v_1^3$$
$$+ 139.65v_1^2 - 157.94v_1 - v_2)^2 + 1) = 0,$$
$$H_2(f_2, \lambda) = 40\lambda(\lambda - 1) + \exp(\lambda(\lambda - 1)) \ln(0.01(2.5v_1^3 + 10.5v_1^2$$
$$+ 11.8v_1 - 0.43v_2^2 + 2.69v_2^2 - 4.56v_2)^2 + 1) = 0,$$
$$(v_1 - c_1)^2 + (v_2 - c_2)^2 + (\lambda - c_3)^2) - 0.36 = 0. \tag{4.3}$$

The Spherical method is applied to the system of equations and the results are in table

4.1, where reversion is detected using the calculus of normal vector. Then the strategy to

Iter.	x_1	x_2	λ	$angle(x_1)$	$angle(x_2)$	$angle(\lambda)$
1	7.164878945	21.584487168	0.5	102.5/102.5	24.1/24.1	110.3/110.3
2	7.158354005	21.557119274	0.489587336	103.2/77.4	15.9/155.8	98.6/69.6
3	7.151475984135986	21.528269750	0.485070949	103.3/76.7	14.9/164.0	96.6/81.3

Table 4.1: Numerical results of path tracking in backward direction.

avoid reversion problem is applied, obtaining the following results Table 4.2 The Figure

Iter.	x_1	x_2	λ	$angle(x_1)$	$angle(x_2)$	$angle(\lambda)$
1	7.164878945	-21.584487168	0.5	102.5/102.5	24.1/24.1	110.3/110.3
2	7.158354005	-21.557119274	0.510412663	103.2/77.4	15.9/155.8	81.3/110.3
3	7.151475984	-21.528269750	0.514929050	103.3/76.7	14.5/165.0	84.4/96.6

Table 4.2: Numerical results of path tracking in backward direction.

4.2 shows the solutions founded using Spherical method and the methodology to avoid the reversion. As a result of tracing the homotopy path, the nine operation points of the circuit has been located (see Table 4.3) all solutions, iteration number for each solution, and the error using mean squares are shown.
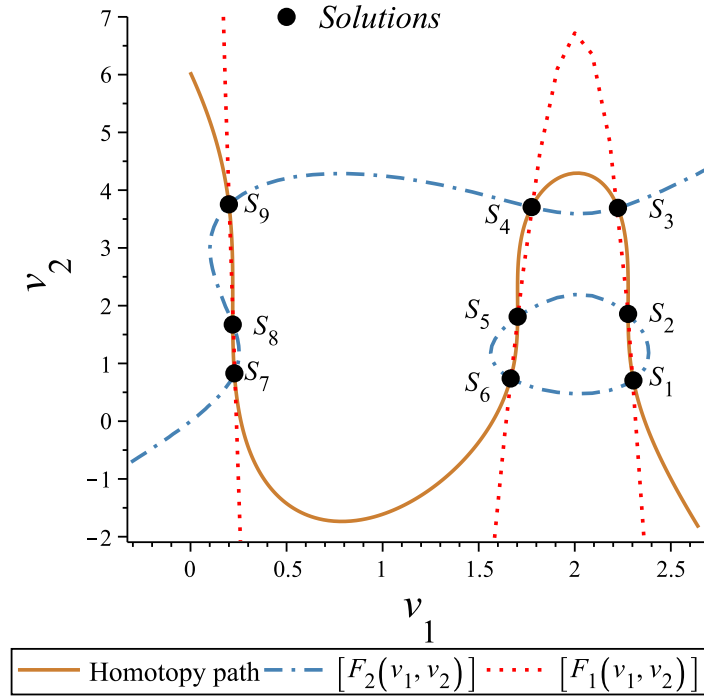
Figure 4.2: Homotopy path for (4.7)

Solution	Iteration	v_1	v_2	$Error = \sqrt{f_1^2 + f_2^2}$
S_1	106	2.305222063	0.705560377	1.3E-10
S_2	143	2.277597006	1.857491731	3.3E-10
S_3	205	2.224729753	3.693043974	6.5E-10
S_4	249	1.775503561	3.707177714	8.8E-11
S_5	316	1.702657758	1.809029946	1.4E-10
S_6	349	1.666377840	0.739343469	5.8E-11
S_7	533	0.228266851	0.828626137	1.2E-11
S_8	558	0.219854573	1.672951409	1.03E-10
S_9	628	0.199790592	3.754217099	1.10E-11

Table 4.3: Numerical solutions to the equation (4.7).

Using a step size of 0.03 and 722 iterations, all solutions has been founded using DBH homotopy. The Figure 4.3 show the variable v_1 versus λ and the Figure 4.4 show the variable v_2 versus λ for the nine solutions.
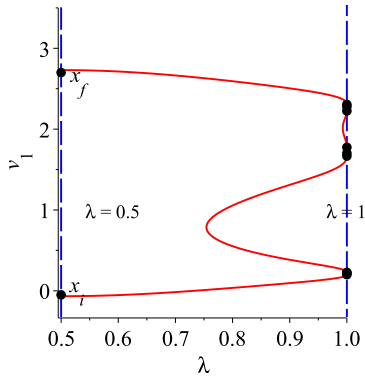
(a) Homotopy path proyected over v_1-λ
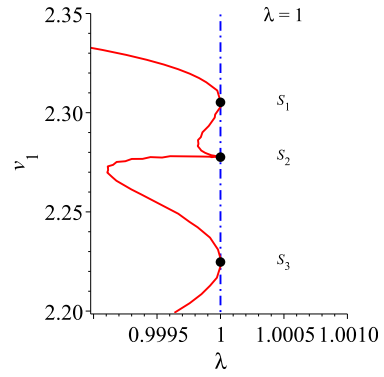
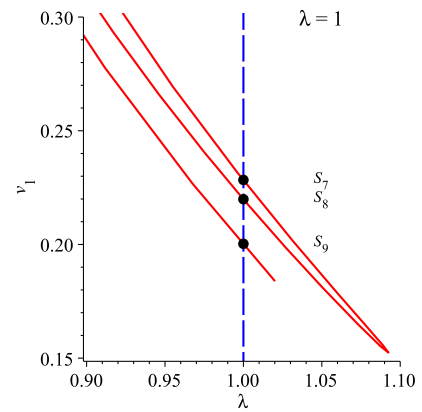(b) Zoom to solutions of (a)
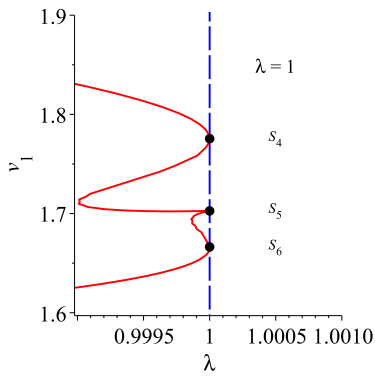
(c) Zoom to solutions of (a)
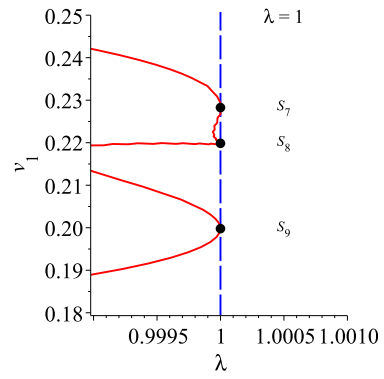
(d) Zoom to solutions of (a)

Figure 4.3: The projection v_1 over λ.

(a) Homotopy path proyected over v_2-λ

(b) Zoom to solutions of (a)
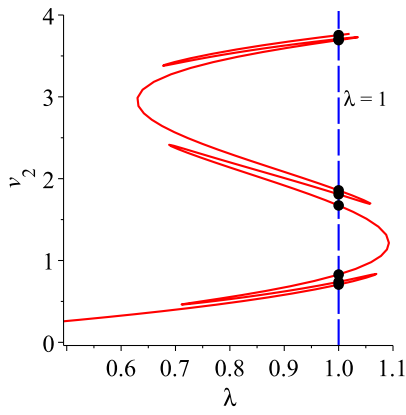

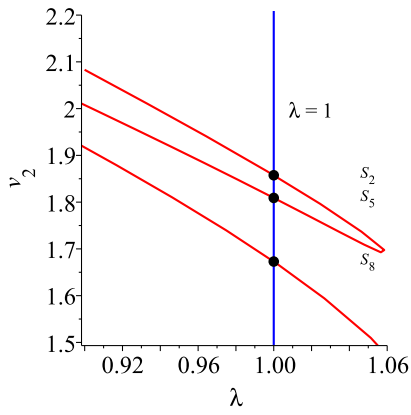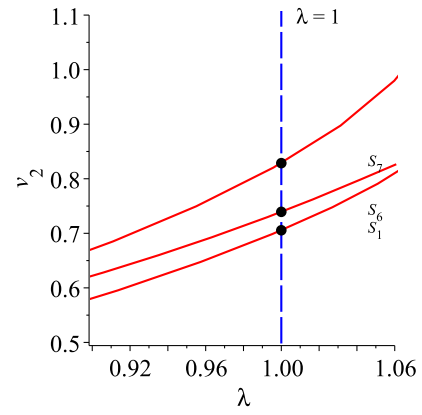(c) Zoom to solutions of (a)

(d) Zoom to solutions of (a)
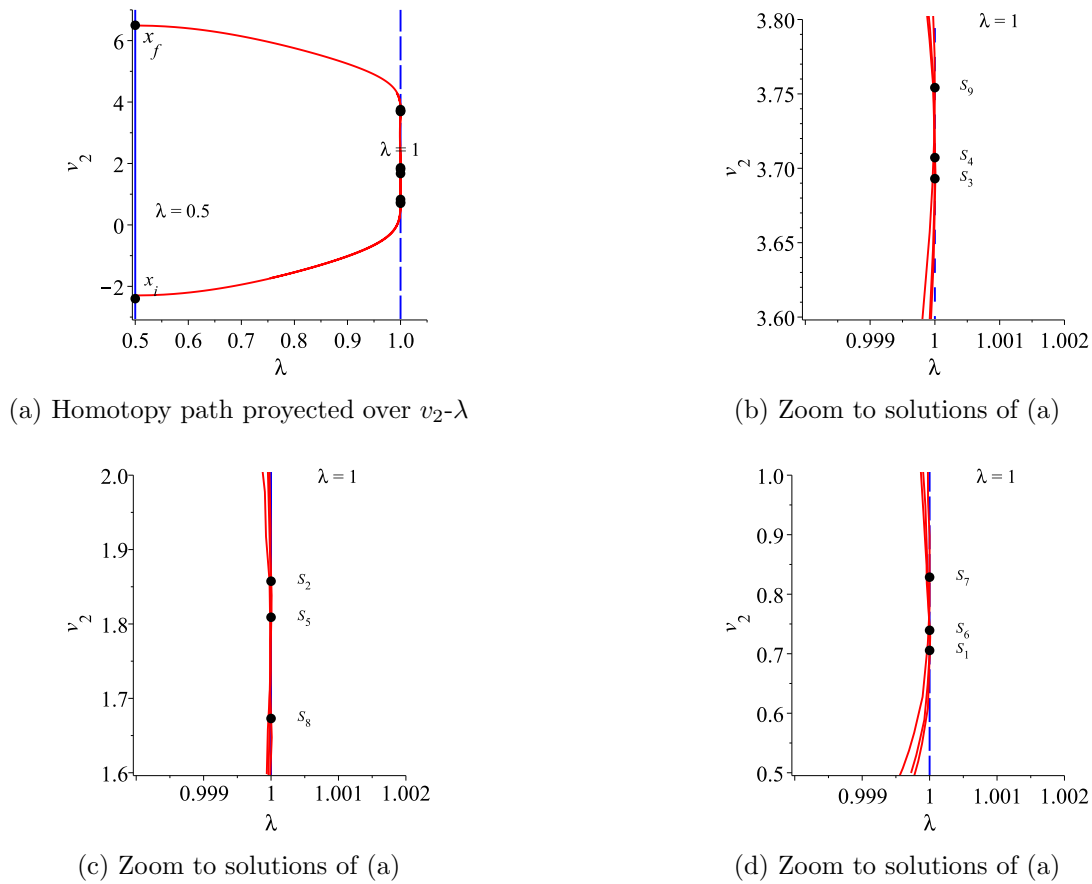
Figure 4.4: The projection v_2 over λ.

As presented in Chapter 2, there are different homotopy functions; each one with advantages and disadvantages for specific case studies. The DBH homotopy show advantages as reliable stop criterion; nonetheless the homotopy formulation require higher number of iterations while Newton homotopy needed a lower number of iterations. Taking into account all above mentioned contributions to methodology for variable radius can be applied in order to reduce the iteration number for traced solution curve. The homotopic formulation used for a system of equations increased $f(x, \lambda)$ is the Newton homotopy as follow.

$$H_1(f_1, \lambda) = 30 - 33.25v_1^3 + 139.65v_1^2 - 157.94v_1 - v_2 - (1-y)f(x_0),$$
$$H_2(f_2, \lambda) = 2.5v_1^3 - 10.5v_2^2 + 11.8v_1 - 0.43v_2^3 + 2.69v_2^2 - 4.56v_2 - (1-y)f(x_0),$$
$$(v_1 - c_1)^2 + (v_2 - c_2)^2 + (\lambda - c_3)^2) - r = 0. \quad (4.4)$$

In order to trace the homotopy path spherical algorithm is applied, the proposed methodology the proposal resizes radius of the sphere with relation to the calculation of the radius

Table 4.4: Numerical results for the first case study

	CPU time	Iteration number
same size radio (0.06)	7.03	205
diferent size radio (0.06-0.1)	6.1	146

of curvature is applied too [4]. A comparison of the simulation results using spheres of fixed radius versus variable radius, see Figure 4.10. The number of iterations is notoriously
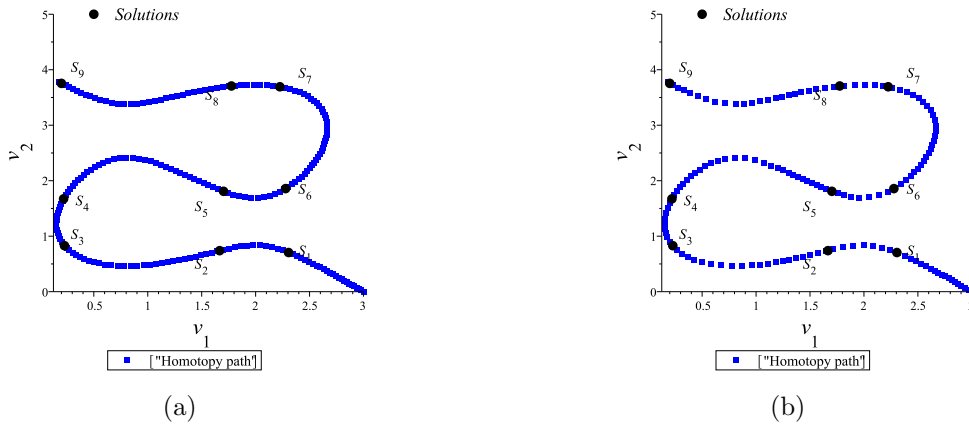
Figure 4.5: (A) Radius size constant. (B) Radius size variable.

reduced and therefore there are a difference in CPU time; using in both cases the Newton homotopy formulation.
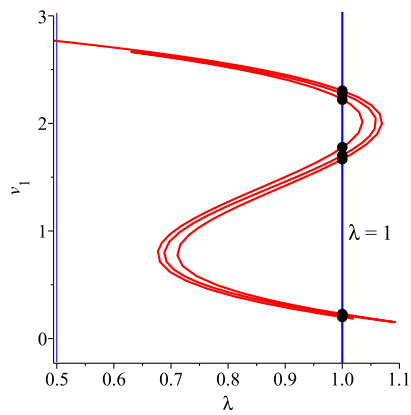
(a) Homotopy path proyected over v_1-λ using Newton Homotopy
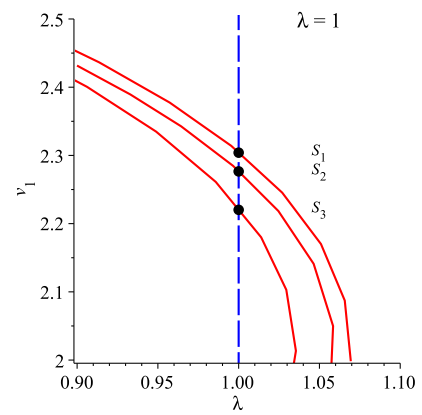
(b) Zoom to solutions of (a)
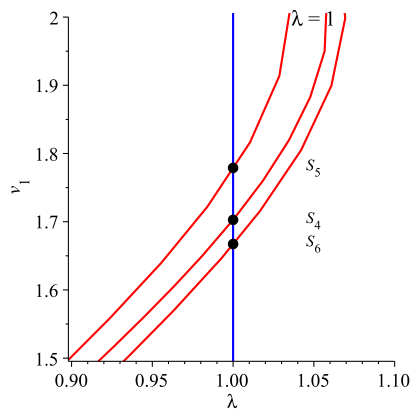
(c) Zoom to solutions of (a)

(d) Zoom to solutions of (a)

Figure 4.6: The projection v_1 over λ using Newton Homotopy.

(a) Homotopy path proyected over v_1-λ using Newton Homotopy
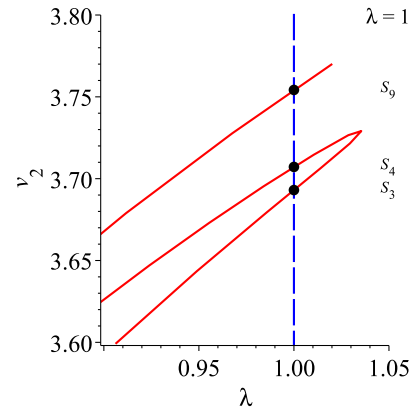
(b) Zoom to solutions of (a)

(c) Zoom to solutions of (a)

(d) Zoom to solutions of (a)

Figure 4.7: The projection v_1 over λ using Newton Homotopy.

4.1.1 Mathematical example

In order to validate the proposed path tracking algorithm, we will solve to the following problem [1]

$$F_1(x_1, x_2) = 4(x_1^2 + x_2^2 - 1)x_1 + 16((2x_1^2 - 1)^2$$
$$+ (2x_2^2 - 1)^2 - 2/3)(2x_1^2 - 1)x_1 = 0,$$
$$F_2(x_1, x_2) = 4(x_1^2 + x_2^2 - 1)x_2 + 16((2x_1^2 - 1)^2$$
$$+ (2x_2^2 - 1)^2 - 2/3)(2x_2^2 - 1)x_2 = 0. \quad (4.5)$$

Applying the DBH homotopy to (4.5), we obtain the following system of equations

$$H_1(f_1, \lambda) = \lambda(\lambda - 1) + \exp(\lambda(\lambda - 1)) \ln(0.09((4v_1^2 + 4v_2^2 - 4)v_1$$
$$+ 16(2v_1^2 - 1)^2 + 16(2v_2^2 - 1)^2 - 32/3)(2v_1^2 - 1)v1)^2 + 1) = 0,$$
$$H_2(f_2, \lambda) = \lambda(\lambda - 1) + \exp(\lambda(\lambda - 1)) \ln(0.09((4v_1^2 + 4v_2^2 - 4)v_2$$
$$+ 16(2v_1^2 - 1)^2 + 16(2v_2^2 - 1)^2 - 32/3)(2v_2^2 - 1)v2)^2 + 1) = 0,$$
$$(v_1 - c_1)^2 + (v_2 - c_2)^2 + (\lambda - c_3)^2) = 0.0009. \quad (4.6)$$

where last equation correspond to the equation of the sphere. The parameters homotopy $a = 0$, $b = 1$, $C = 3$, $D = 0.03$ and the ratio size is $r = 0.03$ using DBH homotopy. Without a methodology for avoiding the reversion the results are shown in Table 4.5, where in the first iteration $\lambda = 0.5$ and the second iteration λ go back to $\lambda = 0.47$.

Iter.	x_1	x_2	λ	$angle(x_1)$	$angle(x_2)$	$angle(\lambda)$
1	0.931862833	-0.931862833	0.5	90.2/90.2	89.7/89.7	179.5/179.5
2	0.931707573	-0.931707573	0.470000000	89.7/90.8	90.2/89.1	0.41/178
3	0.931242105	-0.931242105	0.440004014	89.1/91.4	90.8/88.	1.25/177.9

Table 4.5: Numerical results of path tracking in backward direction.

Applying the proposed methodology to find the solutions in forward direction, the numerical solutions are shown in Table 4.6, where $\lambda = 0.5$ in the first iteration at the second iteration $\lambda = 0.53$ tracing the trajectory circumvent the reversion phenomenon.

The solutions S_1, S_2, S_3, S_4, S_5, S_6, S_7 found are shown in Figure 4.8 traced with 119 iterations. The Table 4.7 show 7 solutions from a total 25 furthermore of the accuracy and the number of iterations related to the path tracking of (4.6). The 7 solutions are showed Figure 4.9:

Iter.	x_1	x_2	λ	$angle(x_1)$	$angle(x_2)$	$angle(\lambda)$
1	0.931862833	-0.931862833	0.5	90.2/90.2	89.7/89.7	179.5/179.5
2	0.931707573	-0.931707573	0.530000000	89.7/90.8	90.2/89.1	179.5/1.25
3	0.931242105	-0.931242105	0.559995985	91.4/89.1	90.8/88.5	178.7/2.09

Table 4.6: Numerical results of path tracking in forward direction
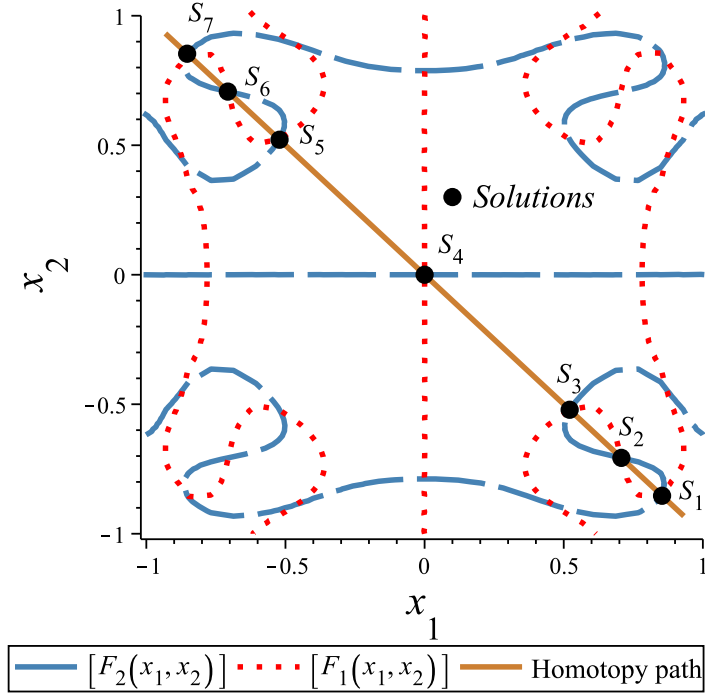
Figure 4.8: Homotopy path for (4.6)

Now the Newton homotopy is applied to (4.7),

$$H_1(f_1, \lambda) = (4v_{1^2}+4v_2^2-4)v_1+(16(2v_1^2-1)^2+16(2v_2^2-1)^2-32/3)(2v_1^2-1)v_1-(1-\lambda)f(x_0)$$
$$H_2(f_2, \lambda) = (4v_1^2+4v_2^2-4)v_2+(16(2v_1^2-1)^2+16(2v_2^2-1)^2-32/3)(2v_2^2-1)v_2-(1-\lambda)f(x_0),$$
$$(v_1 - c_1)^2 + (v_2 - c_2)^2 + (\lambda - c_3)^2) - r = 0. \quad (4.7)$$

Next the methodology for spheres of variable radius is applied, the results of simulation seen in Figure 4.10 indicate the comparison from fixed radius spheres versus variable radius spheres respectively . The iteration number is improved in the case of the variation in size of radius for the sphere in the spherical tracing method applied to the path homotopic. The projected solutions regarding λ parameter are shown Figure 4.11; it shows that points to the first figure are closest with respect at the where size is not fixed radius. Must be emphasized that the solutions have symmetric values and figures only appear 12 repeated

Solution	Iteration	v_1	v_2	$Error = \sqrt{f_1^2 + f_2^2}$
S_1	19	0.853356743	-.853356743	3E-15
S_2	26	0.707106781	-0.707106781	8.61E-14
S_3	34	0.521327409	-0.521327409	3.29E-11
S_4	61	-1.71E-16	1.70E-16	2.4E-11
S_5	87	-0.521327409	0.521327409	2E-15
S_6	96	-0.707106785	0.707106785	2E-15
S_7	103	-0.853356744	0.853356744	9.6E-11

Table 4.7: Numerical solutions to the equation (4.6)

solutions. The table 4.8 shows the results with all the solutions. The comparison of results

Solution	Iteration	x_1	x_2
S_1	27	-0.457846075	-0.88583237
S_2	36	-0.52105087	-0.52233449
S_3	42	-0.706918715	-0.705518887
S_4	46	-0.856403500	-0.840392018
S_5	52	-0.887998886	-0.459719460
S_6	60	-0.788768474	0.000686285
S_7	67	-0.887951404	0.459867289
S_8	74	-0.853645422	0.851469150
S_9	78	-0.706365080	0.703951513
S_{10}	83	-0.521087102	0.522784769
S_{11}	90	-0.459290981	0.887511352
S_{12}	107	0.000467550	0.783519531
S_{13}	117	-0.000003365	-0.000071349
S_{14}	126	-0.000102981	-0.785701888
S_{15}	143	0.457897013	-0.885735694
S_{16}	150	0.521107681	-0.52255104
S_{17}	156	0.706976322	-0.70475460
S_{18}	160	0.854672950	-0.845544717
S_{19}	166	0.888023651	-0.459737787
S_{20}	174	0.788560647	0.001018800
S_{21}	180	0.887897377	0.459769872
S_{22}	187	0.854728259	0.847735395
S_{23}	191	0.706625443	0.705505264
S_{24}	195	0.521211842	0.521668777
S_{25}	204	0.457859208	0.885858576

Table 4.8: Numerical solutions to the equation (4.6)
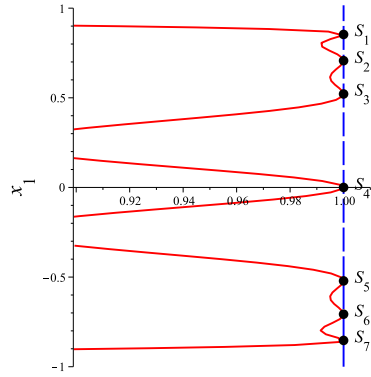
(a)
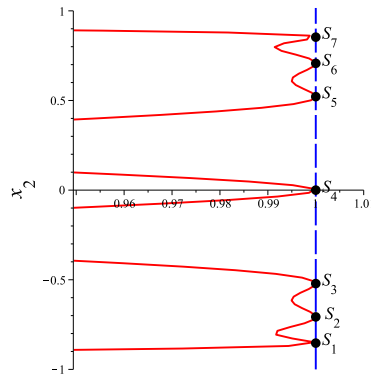
(b)

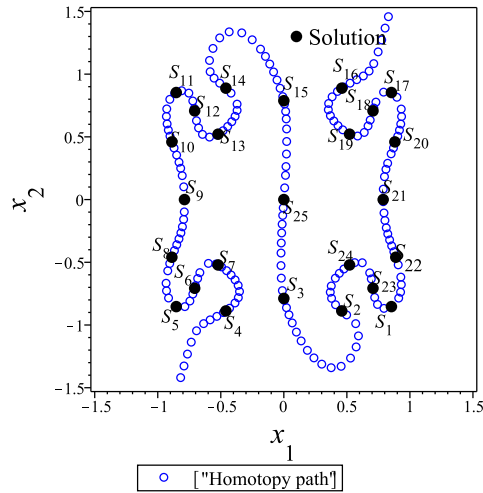Figure 4.9: (a) Radius size constant. (b) Radius size variable.

fixed-radius versus variable-radius are showed table 4.9 where the iteration number has reduced and consequently has happened the same with the computation time.

Table 4.9: Numerical results for the first case study

	CPU time	Iteration number
fixed radius (0.03)	7.4	205
variable radius (0.03-0.1)	14.3	462
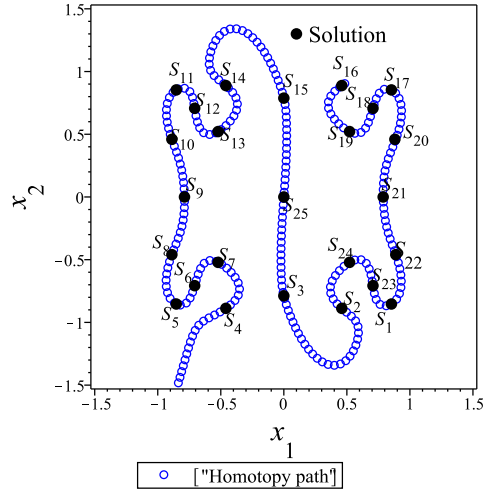
(a)

(b)

Figure 4.10: Projection of x_1 versus x_2.

4.1.2 Circuit with two tunnel exponential diodes

For the diode circuit Figure 4.1 the polynomial expression is replaced by exponential terms

$$i_{1,2} = I_p(\frac{V}{V_p})e^{1-\frac{V}{V_p}} + I_0 e^{\frac{q}{KT}V} \tag{4.8}$$

where $E = 1$, $R = 20\Omega$, $I_p = 100E - 03$, $V_p = 50E - 03$, $I_0 = 1E - 09$ and $\frac{q}{KT} = 40$. As a result the path tracking are show the projection of I_{v_1}, v_2, v_3 over λ. The Figure 4.13 show that the method has found 5 five solutions into the homotopy path.
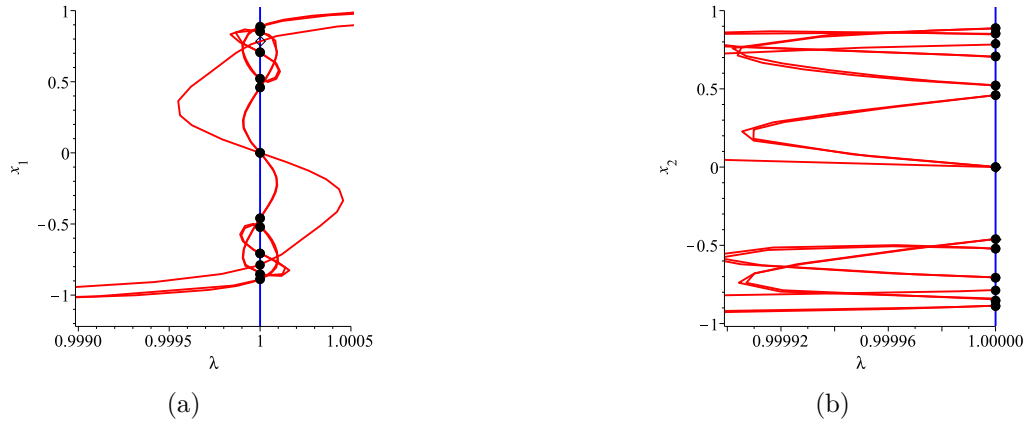
Figure 4.11: (a) x_1 versus λ. (b)x_2 versus λ.

The 5 solutions are showed in table 4.10. Furthermore Newton homotopy function

$Solution$	$Iteration$	I_{v_1}	v_2	v_3	$Error = \sqrt{f_1^2 + f_2^2}$
S_1	45	-0.048873941	0.022521177	0.011260588	1.1E-13
S_2	52	-0.042175710	0.156485795	0.147132103	6.5E-10
S_3	70	-0.018919917	0.621601653	0.418182617	1.5E-11
S_4	77	-0.028280221	0.434395575	0.428548398	2.4E-11
S_5	80	-0.009914377	0.801712456	0.400856225	2.3E-09

Table 4.10: Numerical solutions for the circuit Figure 4.1 with exponential terms.

is applied to find the solutions Figure 4.1 using diode functions with exponential terms, where all solutions are founded. The projection of I_{v1}, v_2 and v_3 versus λ
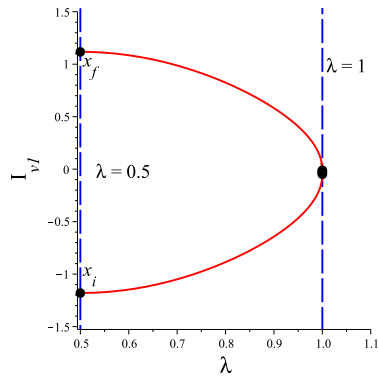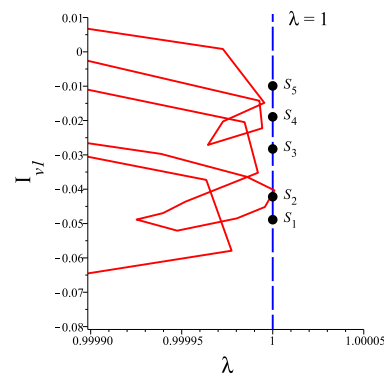
(a) Homotopy path proyected over I_{v_1}-λ

(b) Zoom to solutions of (a)
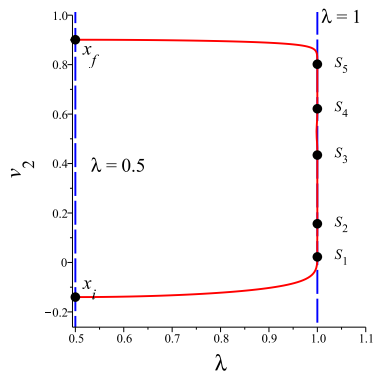
(c) Homotopy path proyected over v_2-λ
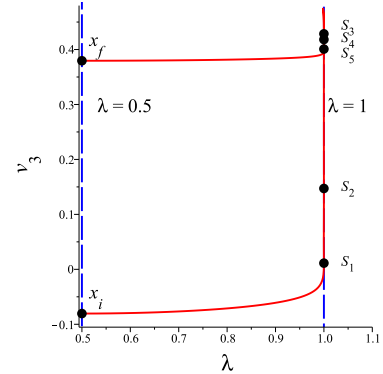
(d) Homotopy path proyected over v_3-λ.

Figure 4.12: The projection I_{v_1}, v_2, v_3 over λ.

Using the method with strategy for variable radio could find all solutions with fewer iterations where all solutions were shown Table 4.11. The solutions found using the
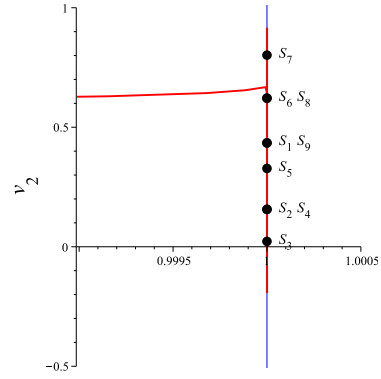
Solution	Iteration	I_{v1}	v_2	v_3
S_1	32	-0.028280220	0.434395581	0.005847176
S_2	36	-0.042175710	0.156485796	0.009353691
S_3	39	-0.048873941	0.022521177	0.011260588
S_4	47	-0.042175710	0.156485796	0.147132104
S_5	52	-0.033613653	0.327726928	0.163863464
S_6	55	-0.018919917	0.621601656	.203419036
S_7	84	-0.009914377	0.801712451	0.400856225
S_8	87	-0.018919917	0.621601653	0.418182617
S_9	92	-0.028280221	0.434395574	0.428548398

Table 4.11: Numerical solutions to the equation (4.8)

methodology of variable Table 4.12, the results show fewer iterations and consequently
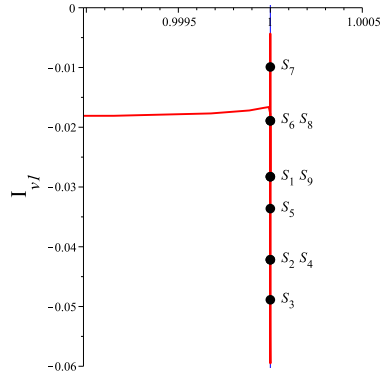
(a) Homotopy path proyected over I_{v_1}-λ

(b) Zoom to solutions of (a)
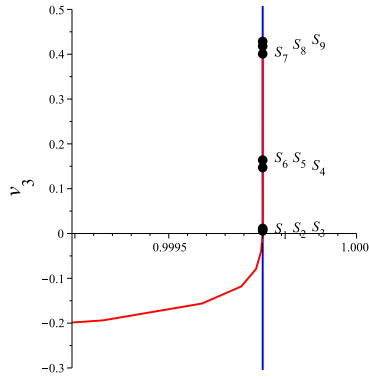
(c) Homotopy path proyected over v_2-λ

Figure 4.13: The projection $I_{I_{v1}}$, v_2, v_3 over λ.

computation time is also decreased. Obtaining a great advantage of the methodology of variable radius on fixed radius using the spherical algorithm for tracing the curve.

Table 4.12: Solutions third case study

Solution	Iter.(r=0.03)	CPU time (sec.)	Iter.(r=0.03-0.1)	CPU time (sec.)
$S_1, S_2, \ldots, S_9$	110	7.1	92	6.6

4.1.3 Four-transistor multistate circuit

The next circuit contain four-transistor shown Figure 4.14 Bipolar transistor is replaced
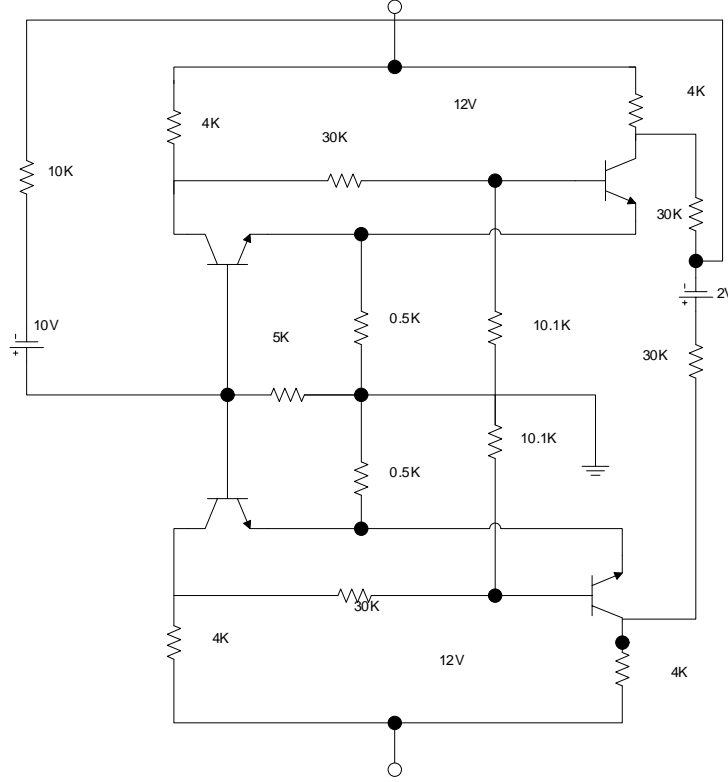
Figure 4.14: A four-transistor multistate circuit

by the simplified EbersMoll model Figure 4.15.

Replacing each bipolar-junction transistor by simplified EbersMoll model [1], obtain four nonlinear equations involving the four unknown base- emitter voltages v_1, v_2, v_3, v_4 where $I_0 = 10E - 06$.

$$f_1 = 6.103168 I_0(\exp(40v_1) - 1) + 4.36634v_2 + 2.863168(\exp(40v_2) - 1) - 12$$
$$f_2 = 5.4v_1 + 3.58 I_0(\exp(40v_1) - 1) + 6.62 I_0(\exp(40v_2) - 1) + v_3$$
$$+ 0.7 I_0(\exp(40v_3) - 1) + 0.5 I_0(\exp(40v_4) - 1) - 22$$
$$f_3 = 6.103168 I_0(\exp(40v_3) - 1) + 2.863168 I_0(\exp(40v_4) - 1) + 4.36634v_4 - 12$$
$$f_4 = 1v_1 + 0.7 I_0(\exp(40v_1) - 1) + 0.5 I_0(\exp(40v_2) - 1) + 5, 4v_3$$
$$+ 3.58 I_0(\exp(40v_3) - 1) + 6.62 I_0(\exp(40v_4) - 1) - 20 \quad (4.9)$$
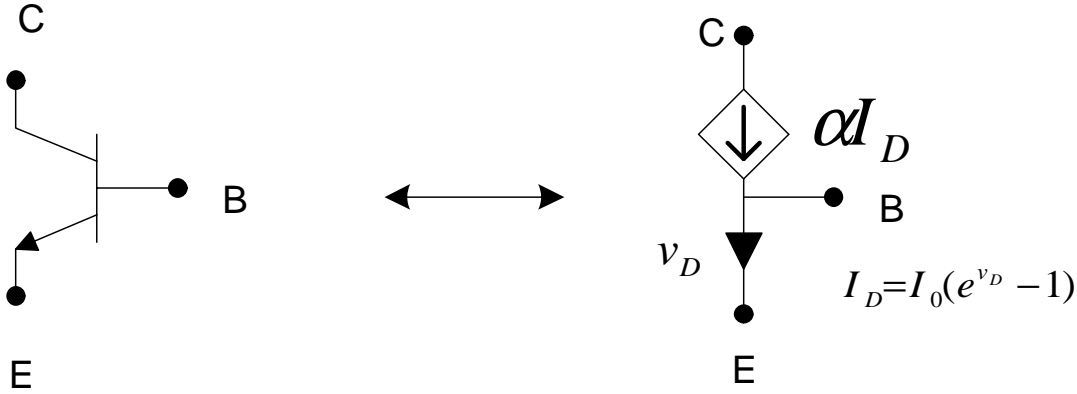
Figure 4.15: Ebers-Moll model

Applying homotopy function is obtained:

$$H_1(f_1, \lambda) = (6.10exp(-06)(exp(40v_1) - 1) + 4.36v_2 + 2.86exp(-06)$$
$$(exp(40v_2) - 1) - 12 + (\lambda - 1)(-6.10exp(-06) - 12)$$
$$H_2(f_2, \lambda) = 6.10exp(-06)(exp(40v_3) - 1) + 2.86exp(-06)(exp(40v_4) - 1)$$
$$+ 4.36v_4 - 12 + (\lambda - 1)(-14.18 - 2.86exp(-06))$$
$$H_3(f_3, \lambda) = v_1 + 0.7exp(-06)(exp(40v1) - 1) + 0.5exp(-06)(exp(40v_2) - 1)$$
$$+5.4v_3+3.58exp(-06)(exp(40v_3)-1)+6.62exp(-6)(exp(40v_4)-1)-20+(\lambda-1)(-20.5-7.31exp(-06))$$
$$H_4(f_3, \lambda) = 5.4v_1 + 3.58exp(-06)(exp(40v_1) - 1) + 6.62exp(-6)(exp(40v_2) - 1)$$
$$+v_3+0.7exp(-06)(exp(40v_3)-1)+0.5exp(-06)*(exp(40v_4)-1)-22+(\lambda-1)*(-24.70-4.079exp(-06))$$
$$H_5(f_3, \lambda) = (v_1 + c_1)^2 + (v_2 - c_2)^2 + (v_3 - c_3)^2 + (v_4 + \lambda)^2 + \lambda^2 - 0.1e - 01 \quad (4.10)$$

For this study case has been applied the Newton homotopy and using and arbitrary initial point the improved Spherical Algorithm is used to trace the path. The table 4.13 shows the solutions found in different paths with solution number, initial point, iteration number using fixed radius, iteration number using variable radius and solutions. The results show

No.	$Initial - point$	(0.03)	(0.03-0.1)	Solution
S_1	(-0.5,0,0,-0.5)	138	56	(-1.42801288,0.18413951,0.19101314,-4.45638520
S_2	(-0.5,-0.5,-0.5,-0.5)	204	80	(0.19011567,-4.201978519,0.186391138,-3.239254017)
S_3	(0.5,0.5,0.50.5)	42	18	(0.13548515,0.17382002,0.14379723,0.16926345)
S_5	(-1,0,-1,0)	35	15	(-0.84596284,0.18413802,-1.30050829, 0.18413802)
S_5	(0,0,-1,0)	183	73	(0.19411919, -5.41008207, -1.88740370, 0.18413976)
S_6	(0.1,0.1,-0.1-0.1)	134	55	(0.14428324,0.17020052,0.18921466,-3.95577056)
S_7	(-0.1,-0.1,0.1,0.1)	160	65	(0.19204522, -0.19204522, 0.15082057,0.16393479)

Table 4.13: Numerical solutions to Four-Transistor Multistate Circuit

benefits in the number of iterations and the computation time for the variable radius against the fixed radius of the sphere, using the spherical algorithm for the tracing of solution curve.

4.1.4 Example with 14 variables

The following case study is the diode-transistor circuit that contain four transistor and one diode as shown in Figure 4.16, where the diodes have an exponential function.
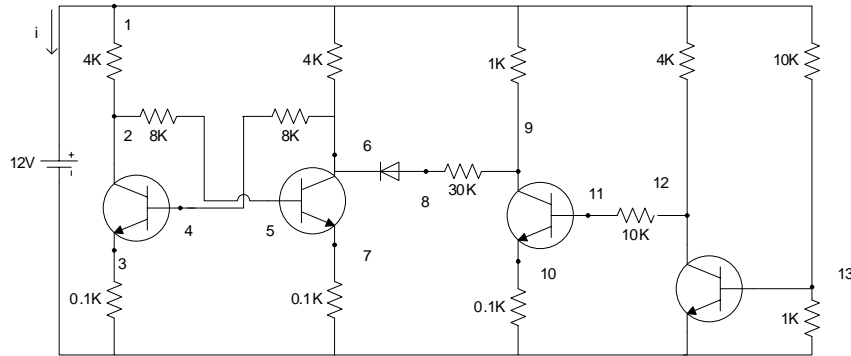

Figure 4.16: Example circuit with 14 variables.

Bipolar transistor is replaced by Ebers-Moll model Figure 4.17.
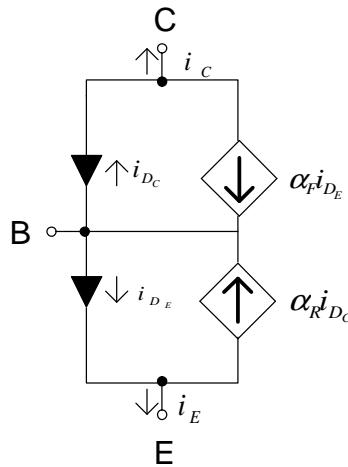
Figure 4.17: Ebers-Moll model used in example circuit with 14 variables.

The Ebers-Moll model is defined by

$$
\begin{bmatrix} i_e \\ i_c \end{bmatrix} = \begin{bmatrix} 1 & -0.01 \\ -0.99 & 1 \end{bmatrix} \begin{bmatrix} 10^{-9}(e^{40v_{be}} - 1) \\ 10^{-9}(e^{40v_{bc}} - 1) \end{bmatrix}
$$

The equilibrium equations are obtained from the modified nodal analysis by:

$f_1 = \frac{37}{20000}v_1 - \frac{1}{4000}v_2 - \frac{1}{4000}v_6 - \frac{1}{1000}v_9 - \frac{1}{4000}v_{12} - \frac{1}{10000}v_{13} + I_E$

$f_2 = -\frac{1}{40000}v_1 + \frac{3}{8000}v_2 + \frac{3}{8000}v_5 + 9.90E - 09\exp^{40v_4 - 40v_3} + 1E - 10 - 1E - 08\exp^{40v_4 - 40v_2}$

$f_3 = \frac{1}{100}v_3 - 1E - 08\exp^{40v_4 - 40v_3} + 9.90E - 09 + 1E - 10\exp^{40v_4 - 40v_2}$

$f_4 = \frac{1}{8000}v_4 - \frac{1}{8000}v_6 + 1E - 10\exp^{40v_4 - 40v_3} - 1E - 08 + 9.90E - 09\exp^{40v_4 - 40v_2}$

$f_5 = \frac{1}{8000}v_2 + \frac{1}{8000}v_5 + 1E - 10\exp^{40v_5 - 40v_7} - 1E - 08 + 9.90E - 09\exp^{40v_5 - 40v_6}$

$f_6 = \frac{1}{4000}v_1 - \frac{1}{8000}v_4 + \frac{3}{8000}v_6 + 9.90E - 09\exp^{40v_5 - 40v_7} + 1.010E - 08\exp^{40v_5 - 40v_6} - 1E - 08\exp^{40v_8 - 40v_6}$

$f_7 = \frac{1}{100}v_7 - 1E - 08\exp^{40v_5 - 40v_7} + 9.90E - 09 + 1E - 10\exp^{40v_5 - 40v_6}$

$f_8 = \frac{1}{30000}v_8 - \frac{1}{30000}v_9 + 1E - 08\exp^{40v_8 - 40v_6} - 1.E - 08$

$f_9 = \frac{1}{1000}v_1 - \frac{1}{30000}v_8 + \frac{31}{30000}v_9 + 9.90E - 09\exp^{40v_{11} - 40v_{10}} + 1E - 10 + 1E - 08\exp^{40v_{11} - 40v_9}$

$f_{10} = \frac{1}{100}v_{10} - 1E - 08\exp^{40v_{11} - 40v_{10}} + 9.90E - 09 + 1E - 10\exp^{40v_{11} - 40v_9}$

$f_{11} = \frac{1}{10000}v_{11} - \frac{1}{10000}v_{12} + 1e - 10\exp^{40v_{11} - 40v_{10}} - 1E - 08 + 9.9E - 09\exp^{40v_{11} - 40v_9}$

$f_{12} = -\frac{1}{4000}v_1 - \frac{1}{10000}v_{11} + \frac{7}{20000}v_{12} + 9.9E - 12\exp^{40v_{13}} + 1E - 10 - 1E - 08\exp^{40v_{13} - 40v_{12}}$

$f_{13} = -\frac{1}{10000} + \frac{11}{10000}v_{13} + 1E - 10\exp^{40v_{13}} - 1E - 08 + 9.90E - 09\exp^{40v_{13} - 40v_{12}}$

$f_{14} = v_1 - 12,$

$$(4.11)$$

Applying the Newton homotopy and going through the curve using the methodology for variable radius spherical method we obtain the solution in Table 4.17.

The results fixed radius versus variable radius are showed in table 4.15.

The results indicate a considerable decrease in the number of iterations with respect to the use of a fixed radius for each sphere. Notwithstanding a path connecting the solutions will not be found and trajectories were taken with a solution While the CPU time also is reduced for fixed radius and variable radius respectively. The projection of variable versus the λ parameter can be plotted, Figure 4.18 shows I_E versus λ and v_{13} versus λ.

Table 4.14: Solutions for example with 14 variables

Sol	S_1	S_2	S_3
I_E	-0.0100	-0.0089	-0.0086
v_1	12	12	12
v_2	0.8839	5.9958	0.4057
v_3	0.2775	0.0849	0.3655
v_4	0.5908	0.3686	0.6858
v_5	0.6318	0.7126	0.3494
v_6	0.8128	0.4364	6.7966
v_7	0.3153	0.3907	0.0704
v_8	1.0745	0.6990	7.0386
v_9	11.6475	0.6354	11.8399
v_{10}	0.0000	0.0000	0.0000
v_{11}	0.0388	0.0388	0.0388
v_{12}	0.0387	0.0387	0.0387
v_{13}	0.3214	0.3214	0.3214

Table 4.15: Solutions fixed radio-variable radio

Initial point	Sol.	Iter.(r=0.03)	CPU time(sec)	Iter.(r=0.03-0.1)	CPU time(sec)
(0, 10, 10, 10, 10, 0, 10, 10, 10, 10, 10, 10, 10, 10)	S_1	478	696.2	180	601.7
(0, 12, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0)	S_2	478	696.2	172	601.7
(0.5, 0.5, 0.5, 0.5, 0.5, 0.5,0.5, 0.5, 0.5, 0.5, 0.5, 0.5, 0.5)	S_3	476	696.2	171	601.7
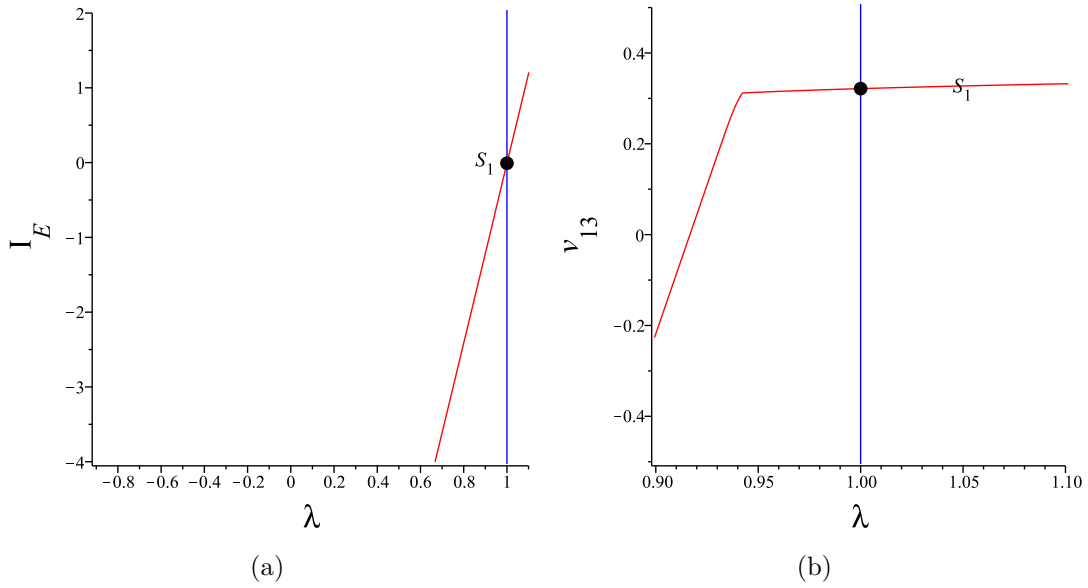
Figure 4.18: (a) Radius size constant. (b) Radius size variable.

Table 4.16: Result Multistate Circuit fixed radio-variable radio

Solution	Iter.(r=0.03)	CPU time (sec.)	Iter.(r=0.03-0.1)	CPU time (sec.)
S_1	2545	5118.003	865	1835.6

4.1.5   Multistate Circuit

We finally present a case study containing an increased number of transistors and diodes resistors regarding above case studies; the Figure 4.19 shows the circuit for solving. The Ebers Moll model used for this circuit is above Figure 4.17 and the diode model is defined by

$$i_d = 10^{-9}(e^{40u} - 1). \tag{4.12}$$

After obtaining the system of equations we proceed to give a solution using the methodology proposed in this work. We choose a starting point $(-15, -15, -15, -15, -15, -15,$ $-15, -15, -15, -15, -15, -15, -15, -15, -15, -15, -15)$ obtaining a path with a solution.

Iteration number and CPU time are showed in Table 4.16, the solution founded is in Table **??**.

Finally the results obtained for this case study indicates a decrease of the number of iterations using the proposed methodology for present work Figure 4.20, 4.21, 4.22 .

Table 4.17: Solutions Multistate Circuit

Sol	S_1	
I_E	−0.0347	
v_1	12	
v_2	8.2950	
v_3	8.0239	
v_4	0.2842	
v_5	0.3276	
v_6	0.5148	
v_7	0.5148	
v_8	0.7824	
v_9	1.0244	
v_{10}	5.1209	
v_{11}	0.6756	
v_{12}	0.9813	
v_{13}	0.7824	
v_{14}	1.0244	
v_{15}	5.1209	
v_{16}	0.6756	
v_{17}	0.9813	
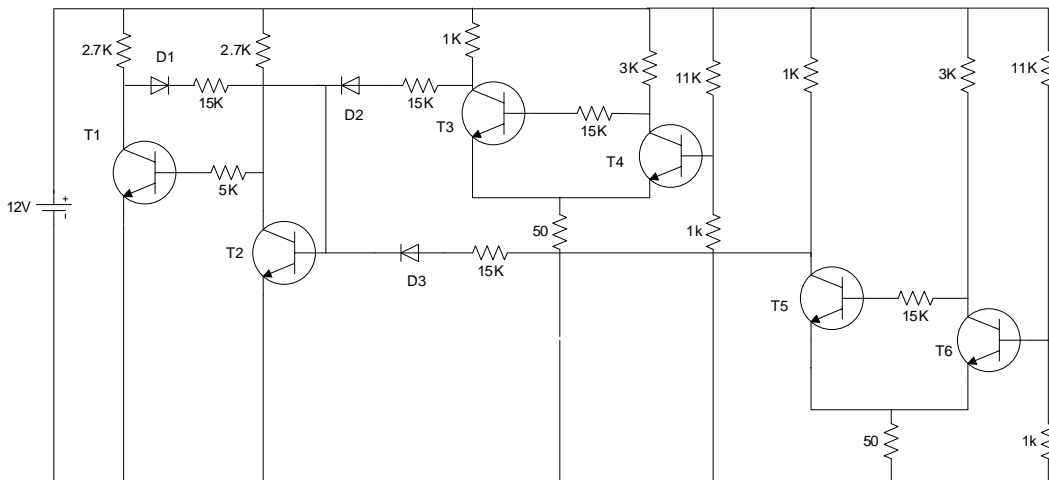
Figure 4.19: Multistate Circuit

4.2 Results and Discussion

The Spherical algorithm has been used with modifications to avoided the reversion problem during steps predictor-corrector for the path tracking. The case studies presented in this chapter have demonstrated that the problem of reversion has been successful detected and avoided. The above has prevented the errors when the curve is traced in the direction backwards further disturbance to the method of tracing has caused the corrector step follow the path on the homotopy curve. The examples 1 and 2 shows the calculus of the angle of normal vector, taking advantage of that numerically the angles are totally different for each solution (forward and backward). In all examples is used the methodology to avoid the reversion problem to the traced of solution curve. The double bounded homotopy and Newton homotopy were used, where the first the stop criterium is a great advantage to decide the number of iterations. Moreover Newton homotopy has the characteristic of connecting the solutions using fewer iterations and diverges when it has already come to the path containing solutions. In general, all homotopy formulation needs a method of tracing, where the path is numeric traced and depends of the step size which is already defined in the beginning to the conventional Spherical Method. Step size has numerical relation to the size of the sphere radius, which is set to a fixed value. In the case studies presented in this chapter for tracing the curve is used variable radius of the spheres, where previously it is possible to know the behavior of the curve using the radius of curvature. To obtain radius of curvature is calculated numerical differentiation using iteration points for a current iteration and a subsequent iteration, where CPU time for such numerical calculations is not significant as can be seen in Table 4.4, Table**??**. The results are successful with respect to tracing of the path solution whereas the number of
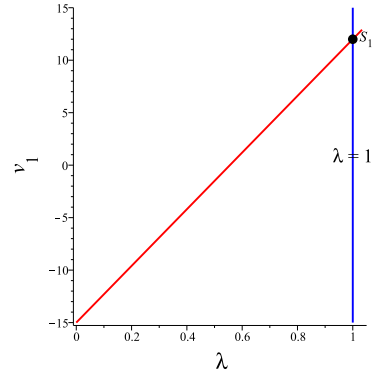
iterations is reduced and the CPU time is less than the radius fixed for the spheres.
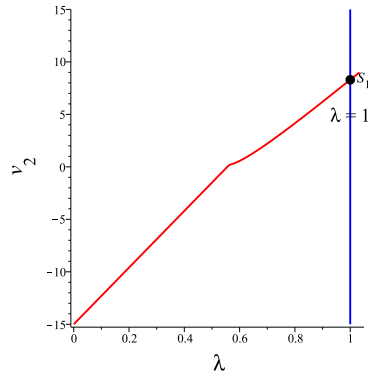
4.3 Summary

In this chapter different study cases show correct detection and circumvent the reversion problem. Furthermore the proposed methodology for implementing a variable radius for spheres resulted successful in reduce the number of iterations for the trace of the trajectory. But nevertheless in the first case studies are presented as case studies to find all solutions of the system; but in the case studies where the number of variables increases the election of starting point increases its complexity. Therefore the absence of an established methodology for choosing the starting point; results in the path trajectories which not contain all solutions. However the methodologies reversion and variable radius have satisfactory performance in all study cases.
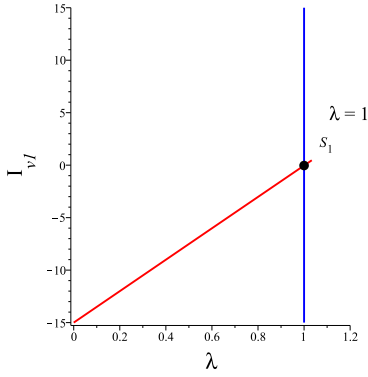
(a) Homotopy path proyected over I_{v_1}-λ

(b) Homotopy path proyected over v_1-λ

(c) Homotopy path proyected over v_2-λ
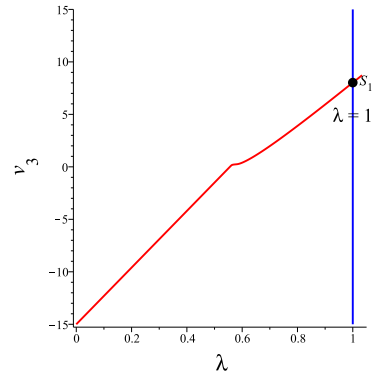
(d) Homotopy path proyected over v_3-λ.
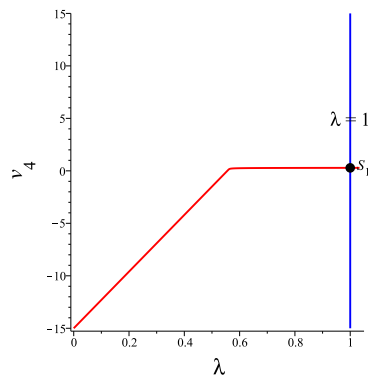
(e) Homotopy path proyected over v_4-λ.
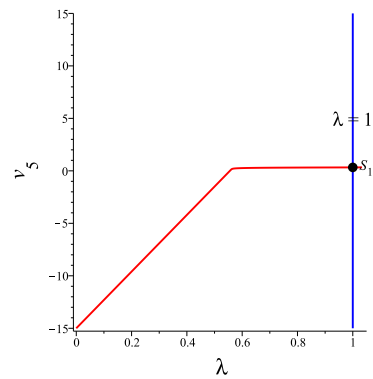
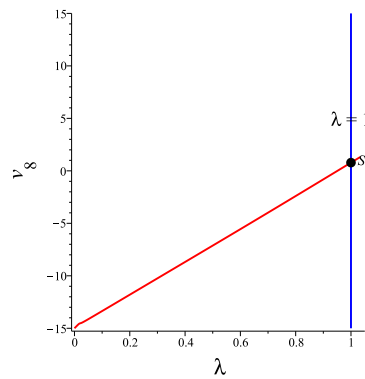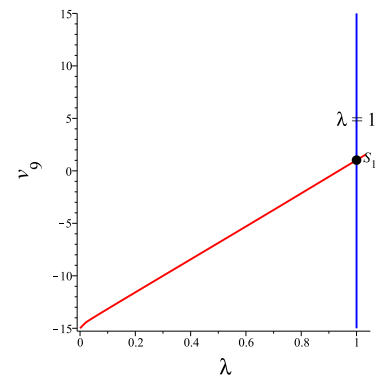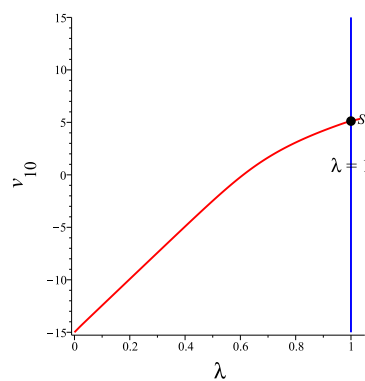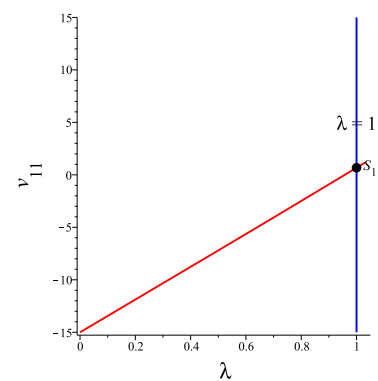(f) Homotopy path proyected over v_5-λ.

Figure 4.20: The projection I_{v_1}, v_1, v_2, v_3, v_4, v_5 , v_5 over λ.
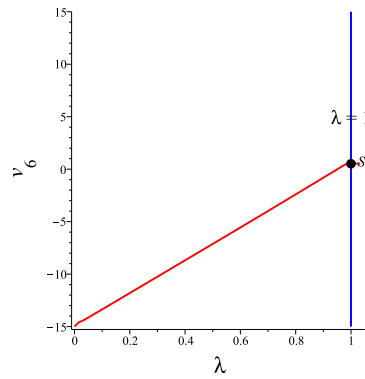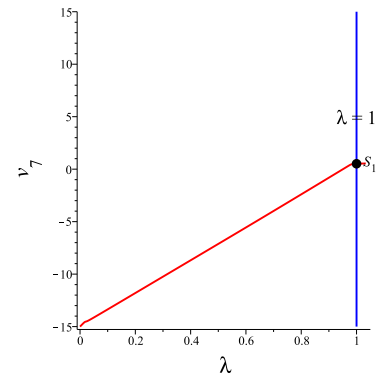
(a) Homotopy path proyected over v_6-λ

(b) Homotopy path proyected over v_7-λ

(c) Homotopy path proyected over v_8-λ

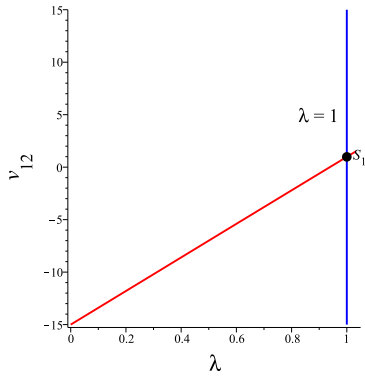(d) Homotopy path proyected over v_9-λ.

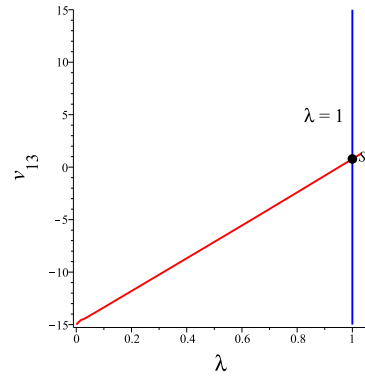(e) Homotopy path proyected over v_{10}-λ.

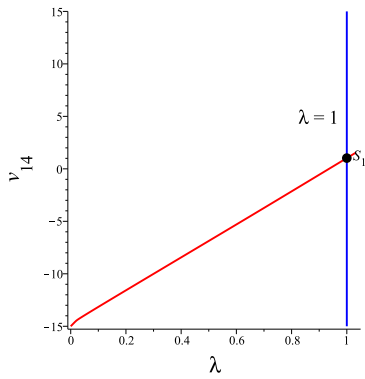(f) Homotopy path proyected over v_{11}-λ.

Figure 4.21: The projection v_6, v_7, v_8, v_9, v_{10}, v_{11} over λ.
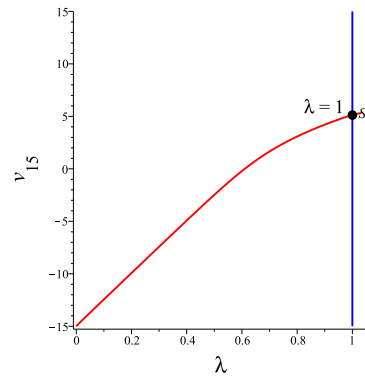
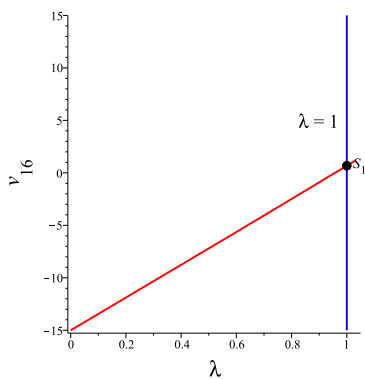(a) Homotopy path proyected over v_{12}-λ
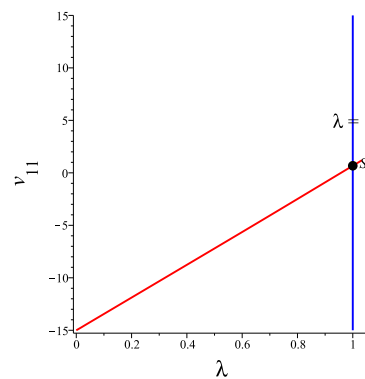
(b) Homotopy path proyected over v_{13}-λ

(c) Homotopy path proyected over v_{14}-λ

(d) Homotopy path proyected over v_{15}-λ.

(e) Homotopy path proyected over v_{16}-λ.

(f) Homotopy path proyected over v_{17}-λ.

Figure 4.22: The projection v_{12}, v_{13}, v_{14}, v_{15}, v_{16}, v_{17} over λ.

Bibliography

[1] J. Lee and H.-D. Chiang. Constructive homotopy methods for finding all or multiple dc operating points of nonlinear circuits and systems. *Circuits and Systems I: Fundamental Theory and Applications, IEEE Transactions on*, 48(1):35–50, 2001.

[2] D. Torres-Munoz, L. Hernandez-Martinez, and H. Vazquez-Leal. Improved spherical continuation algorithm by nonlinear circuit. In *Circuits and Systems (MWSCAS), 2013 IEEE 56th International Midwest Symposium on*, pages 61–64, Aug 2013.

[3] D. Torres-Muoz, H. Vazquez-Leal, L. Hernandez-Martinez, and A. Sarmiento-Reyes. Improved spherical continuation algorithm with application to the double-bounded homotopy (dbh). *Computational and Applied Mathematics*, 33(1):147–161, 2014.

[4] Delia Torres-Muoz, Luis Hernandez-Martinez, and Hector Vazquez-Leal. Spherical continuation algorithm with spheres of variable radius to trace homotopy curves. *International Journal of Applied and Computational Mathematics*, pages 1–13, 2015.

Chapter 5

Conclusions

Spherical algorithm ha been improved and the results support usability applied at study cases with different number of variables. To implement computationally the hypersphere algorithm reversion problem usually occurs during step corrector and causes the curve take a direction opposite to the trajectory. For detecting the problem of reversion the strategy presented is based on the calculation of the normal vector to the curve where the difference between the solution of the left and right is found successfully avoiding numerical problems. Then of the detection of the reversion problem It should implement a strategy to prevent reversion; which it has increased the value of the radius size for the predictor step, order to ensure going forward for tracing the curve. The case studies demonstrate that while the reversion problem is present; using the application of the methodology the problem is detected and avoided successfully. Avoiding the reversion problem the solutions of the equations system can be found on the curve when tracing the homotopy curve where the step size it is fixed to the beginning of the path. If step size is fixed then the traced of the homotopy curve always is traveled using the same radius of the sphere; notwithstanding nonlinearities of the curve are not presented in a constant way such that is possible to assign different radio sizes for each of the spheres depending on the folds that occur in the curve.

To previously know like the curve will behave then calculate the radius of curvature is proposed using numerical differentiation where the radius of curvature calculation does not increase the computation time. In summary to reduce the number of iterations and the computational time the proposed strategy is based on using a different size the spheres radio depending on the folds of the curve using the radius of curvature to propose the size radio. By applying the methodology of variable radius in each case studies presented the results were satisfactory reducing the number of iterations and the computational time.

The case studies used to test the methodologies proposed for this work, they consist of

polynomial equations and exponential type as well as number of variables is different. All study cases reached successful results when compared the number of iterations as well as computation time which are improved. Although the results show that all solutions have not been found, specifically for the latest results; unfortunately parameters as initial point and homotopic formulation were not optimized for each case study.