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

©INAOE 2015
All rights reserved
The author hereby grants to INAOE permission to reproduce and to distribute copies of this thesis document in whole or in part
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 anteriores 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 an 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

<table>
<thead>
<tr>
<th>Chapter</th>
<th>Title</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>Resumen</td>
<td></td>
<td>ii</td>
</tr>
<tr>
<td>Abstract</td>
<td></td>
<td>iv</td>
</tr>
<tr>
<td>1</td>
<td>Introduction</td>
<td>3</td>
</tr>
<tr>
<td>1.1</td>
<td>DC Analysis</td>
<td>3</td>
</tr>
<tr>
<td>1.2</td>
<td>Analysis of Nonlinear Circuits</td>
<td>4</td>
</tr>
<tr>
<td>1.2.1</td>
<td>Analog circuit containing diodes and transistors</td>
<td>5</td>
</tr>
<tr>
<td>1.2.2</td>
<td>Number of DC operating points</td>
<td>6</td>
</tr>
<tr>
<td>1.2.3</td>
<td>Iterative method for analysis of nonlinear circuits</td>
<td>6</td>
</tr>
<tr>
<td>1.2.4</td>
<td>Mathematical representation of electronic devices</td>
<td>7</td>
</tr>
<tr>
<td>1.3</td>
<td>Computer algorithms</td>
<td>8</td>
</tr>
<tr>
<td>1.4</td>
<td>Continuation or homotopy methods</td>
<td>8</td>
</tr>
<tr>
<td>1.5</td>
<td>Motivation</td>
<td>10</td>
</tr>
<tr>
<td>1.6</td>
<td>Objective of the thesis</td>
<td>12</td>
</tr>
<tr>
<td>1.7</td>
<td>Summary</td>
<td>13</td>
</tr>
<tr>
<td>2</td>
<td>Homotopy methods and path tracking</td>
<td>17</td>
</tr>
<tr>
<td>2.1</td>
<td>Formulation a system of equations</td>
<td>17</td>
</tr>
<tr>
<td>2.2</td>
<td>A basic Newton-Raphson method</td>
<td>18</td>
</tr>
<tr>
<td>2.2.1</td>
<td>Dependent techniques of Newton Raphson method</td>
<td>19</td>
</tr>
<tr>
<td>2.3</td>
<td>Homotopy method</td>
<td>21</td>
</tr>
<tr>
<td>2.4</td>
<td>Homotopic formulations</td>
<td>21</td>
</tr>
<tr>
<td>2.4.1</td>
<td>The Fixed-point homotopy</td>
<td>22</td>
</tr>
<tr>
<td>2.4.2</td>
<td>The variable gain homotopy</td>
<td>22</td>
</tr>
<tr>
<td>2.4.3</td>
<td>The Newton homotopy</td>
<td>23</td>
</tr>
<tr>
<td>2.4.4</td>
<td>The Double Bounded Homotopy</td>
<td>23</td>
</tr>
<tr>
<td>2.5</td>
<td>Convergence of homotopy methods</td>
<td>23</td>
</tr>
</tbody>
</table>
## Contents

2.6 Path tracking homotopy ........................................ 25  
2.7 Hyperspherical algorithm for tracing curves .................. 26  
2.8 Find zero strategy ............................................. 27  
2.9 Summary .......................................................... 28  

3 Proposed methodology for the variable radius in the spherical method 33  
3.1 Introduction ....................................................... 33  
3.2 Reversion problem in Spherical algorithm .................... 33  
3.3 Strategy to detect the reversion phenomenon .................. 36  
3.4 Strategy to avoid the reversion phenomenon ................... 38  
3.5 Proposed methodology to accelerate the course of the curve on the spherical algorithm ........................................ 39  
3.6 Aspects to consider in tracing curve ........................... 43  
3.7 Conclusions .......................................................... 44  

4 Case studies 49  
4.1 Non-linear circuit with two tunnel diodes ..................... 50  
4.1.1 Mathematical example ........................................ 58  
4.1.2 Circuit with two tunnel exponential diodes ................ 62  
4.1.3 Four-transistor multistate circuit ............................ 66  
4.1.4 Example with 14 variables .................................... 69  
4.1.5 Multistate Circuit .............................................. 72  
4.2 Results and Discussion ........................................... 74  
4.3 Summary .......................................................... 76  

5 Conclusions 83
# List of Figures

1.1 Ebers-Moll model of a NPN transistor .............................. 5  
1.2 Path tracking of a homotopy curve ................................. 10  
2.1 Geometric interpretation of the Newton Raphson method ........ 18  
2.2 DBH Homotopy .................................................. 24  
2.3 Predictor-corrector scheme ....................................... 25  
2.4 Solution curves with spheres .................................... 27  
2.5 Forward Solution (FS) and Backward Solution (BS) ............ 28  
3.1 Hyperspheres algorithm ........................................... 34  
3.2 Predictor point ................................................... 34  
3.3 Sphere intersecting with the curve at two points ................ 35  
3.4 Solution curves with spheres ................................... 36  
3.5 (A) Forward path. (B) Path back. ................................. 37  
3.6 Normal vector into the sphere ................................... 38  
3.7 Angle to normal vector ......................................... 38  
3.8 Proposed reversion strategy ..................................... 39  
3.9 Numerical differentiation ........................................ 40  
3.10 Hyperbolic tangent function .................................... 41  
3.11 Path tracking of studio case ................................... 43  
3.12 Find zero strategy ............................................... 44  
3.13 Approximate solution ............................................ 45  
4.1 Two tunnel diode circuit .......................................... 50  
4.2 Homotopy path for (4.7) ........................................ 52  
4.3 The projection $v_1$ over $\lambda$ ................................ 53  
4.4 The projection $v_2$ over $\lambda$ ................................ 54  
4.5 (A) Radius size constant. (B) Radius size variable ............ 55  
4.6 The projection $v_1$ over $\lambda$ using Newton Homotopy ....... 56
List of Figures

4.7 The projection $v_1$ over $\lambda$ using Newton Homotopy. .......................... 57
4.8 Homotopy path for (4.6) ................................................................................. 59
4.9 (a) Radius size constant. (b) Radius size variable. ................................. 61
4.10 Projection of $x_1$ versus $x_2$. ................................................................. 62
4.11 (a) $x_1$ versus $\lambda$. (b) $x_2$ versus $\lambda$. ............................................. 63
4.12 The projection $I_{v_1}, v_2, v_3$ over $\lambda$. ............................................. 64
4.13 The projection $I_{v_1}, v_2, v_3$ over $\lambda$. ............................................. 65
4.14 A four-transistor multistate circuit ......................................................... 66
4.15 Ebers-Moll model ..................................................................................... 67
4.16 Example circuit with 14 variables. ........................................................ 69
4.17 Ebers-Moll model used in example circuit with 14 variables. ............... 69
4.18 (a) Radius size constant. (b) Radius size variable. ............................... 72
4.19 Multistate Circuit .................................................................................... 74
4.20 The projection $I_{v_1}, v_1, v_2, v_3, v_4, v_5, v_5$ over $\lambda$. .............. 77
4.21 The projection $v_6, v_7, v_8, v_9, v_{10}, v_{11}$ over $\lambda$. ...................... 78
4.22 The projection $v_{12}, v_{13}, v_{14}, v_{15}, v_{16}, v_{17}$ over $\lambda$. ............... 79
Chapter 1

Introduction

Finding the DC operating points is an 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](image)

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]:

\[ F(x) = 0. \]  

(1.1)

where \( F \) represents the nonlinear expressions and \( x \) is the system variables.
1.2.2 Number of DC operating points

Nonlinear circuits consisting of an arbitrary number of linear resistors 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 points. In general, any circuit that does not possess a unique DC operating point. One circuit that contains more than two transistors may possess numerous operating points. Several methods have 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 and inductors as short circuits. For time domain analysis is based on the choice of discrete time instants $t_n$, for $n = 1, 2, 3,...$; where the input excitation may be an arbitrary function of time. For solving the nonlinear 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 which 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 which 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 possess 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 known 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 process 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 \( \lambda \) 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 aim 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

\[
H(x, \lambda) = 0
\]

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:

\[
F(x, \lambda) = 0,
\]
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

\[
(ds)^2 = (dx_1)^2 + (dx_2)^2 + \ldots + (dx_n)^2
\]  

(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 least 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 correct, 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 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


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 matrix-vector 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:

$$x^{j+1} = x^j - [J(x^j)]^{-1} f(x^j)$$  \hspace{1cm} (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 of the NR method have to be emphasized:
2.2. A basic Newton-Raphson method

- 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 start 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 an 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 the 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 $\alpha$ is initialized to a small number where the increments of $\alpha$ 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 function

\[ F(x) = 0 \quad (2.2) \]

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

\[ H(x, \lambda) \quad (2.3) \]

where \( H : \mathbb{R}^{n+1} \rightarrow \mathbb{R}^n \) and \( x \in \mathbb{R}^n \) 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

\[ \Gamma(H) = (x, \lambda) : H(x, \lambda) = 0 \quad (2.4) \]

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

\[ \Gamma_1(H) = x : H(x, 1) = 0 \quad (2.5) \]

to find all solutions \( \Gamma_0 \) 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 find 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 \( \lambda \) is used as the integration parameter that varies from an initial value \( \lambda_0 \) to a value \( \lambda^* \) where a solution \( x^* \), to the original system is found.

2.4 Homotopic formulations

Several homotopy functions can be constructed from nodal analysis formulated from Kirchhoff laws.
2.4.1 The Fixed-point homotopy

The formulation of fixed point homotopy is expressed as

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

where, \( \lambda \) is a continuation parameter, \( a \) is a random vector and the new parameter \( G \in R^n \times R^m \) 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 \( \lambda \) 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 \( \alpha \) is a vector consisting of transistor forward and reverse current gain. These current gains are multiplied by \( \lambda \). 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 \( \lambda \), the circuit is slowly deformed and the coupling of the transistors p-n junctions is introduced. For each instance of \( \lambda \), 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:

\[ H(x, \lambda) = F(x) - (1 - \lambda)F(x_0), \quad (2.8) \]

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.

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

where \( F(x) \) is the original equation system, \( \lambda \) 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
Chapter 2. Homotopy methods and path tracking

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

$$
H_1(F_1(x), \lambda) = 0,
H_2(F_2(x), \lambda) = 0,
\vdots
H_n(F_n(x), \lambda) = 0,
$$

(2.11)

where (2.11) contain \( n \) number of equations and \( n + 1 \) are number of unknown variables. This causes the system cannot be solved using conventional algebraic methods. The
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 $(x^j, \lambda_j)$ calculated from the equation,

$$ (\bar{x}^{j+1}, \bar{\lambda}^{j+1}) = (x^j, \lambda_j) + h \times t $$

(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 $(x^{j+1}, \lambda_{j+1})$. To path tracking the homotopy curve a starting point is

![Figure 2.3: Predictor-corrector scheme](image)
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:

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

(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 which is situated in the forward direction and another of which 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)
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 be used Hermite cubic.
2.9 Summary

The spherical method to trace the solution curve is simple and geometrically clear. Nonetheless, the 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


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.
Chapter 3. Proposed methodology for the variable radius in the spherical method

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 $\lambda$. 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...
3.2. Reversion problem in Spherical algorithm

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

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

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

$$H_1(F_1(x), \lambda) = 0,$$
$$H_2(F_2(x), \lambda) = 0,$$
$$\vdots$$
$$H_n(F_n(x), \lambda) = 0,$$
$$S(x_1, x_2, \cdots, x_n, \lambda) = 0. \quad (3.2)$$

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 successions 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](image)

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.
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 them. 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)

\[
\nabla S = S'_{x_1} \hat{x}_1 + S'_{x_2} \hat{x}_2 + \cdots + S'_{\lambda} \hat{\lambda}
\]

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.
3.3. Strategy to detect the reversion phenomenon

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

$$\vec{n} = \frac{\nabla S}{\|\nabla S\|}$$  \hspace{1cm} (3.4)

The notation $\|\nabla S\|$ represents the Euclidean norm of $\nabla S$. Then as mentioned previously, it is necessary to obtain the angle of $\vec{n}$ calculated by

$$\theta_{x_i} = \cos^{-1} \frac{S_i'}{\|\nabla S\|}, \quad i = 1, 2, \ldots, n + 1$$  \hspace{1cm} (3.5)

where $\theta_{x_i}$ 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 depend 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.
Chapter 3. Proposed methodology for the variable radius in the spherical method

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 \( \delta r \) 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 \( \delta r \) which coincides with the sphere radius \( r \), where \( \delta r \) is an increase in the radius. The proposed to avoid the reversion problem in the spherical method will be tested later using different
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

\[ \rho = \left| \frac{(1 + (x'_i)^2)^{\frac{3}{2}}}{x''_i} \right| \]  

(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:

\[ x'_i = \frac{x_i - x_{i-1}}{h} \]  

(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).

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

\[ x''_i = \frac{x'_i - x'_{i-1}}{h} \]  

(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 \( x_1, x_2, \ldots x_n \) 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,

\[ \rho_{av} = \frac{\rho_{x_1} + \rho_{x_2} + \cdots + \rho_{x_n}}{n} \]  

(3.9)
3.5. Proposed methodology to accelerate the course of the curve on the spherical algorithm

\[ r = f(\rho_{av}) = \frac{A e^{\rho_{av}} - B e^{-\rho_{av}}}{C e^{\rho_{av}} + D e^{-\rho_{av}}} \] (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, $l_{up}$ and $l_{low}$ 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,

\[
F_1(x_1, x_2) = (4x_1 + 8)e^{-(x_1+2)^2} - x_2^2 + 24x_1e^{-3x_1^2-(x_2-2)^2} + (12x_1 - 36)e^{-2(x_1-3)^2} - x_2^2 + 8x_1e^{-2x_1^2-2(x_2+1)^2} + (6x_1 + 6)e^{-3(x_1+1)^2-3(x_2-2)^2} = 0,
\]

\[
F_2(x_1, x_2) = (8x_2 + 8)e^{-3x_2^2-(x_2-2)^2} + 6x_2e^{-2(x_1-3)^2} - x_2^2 + 4x_2e^{-(x-1+2)^2} + (8x_2 - 16)e^{-3x_2^2-(x_2-2)^2} + (6x_2 - 12)e^{-3(x_1+1)^2-3(x_2-2)^2}.
\] (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).

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

To ensure that the function operates properly to estimate the size of the radius of the
Chapter 3. Proposed methodology for the variable radius in the spherical method

<table>
<thead>
<tr>
<th>Radius of Curvature</th>
<th>radio(r)</th>
</tr>
</thead>
<tbody>
<tr>
<td>0.21</td>
<td>0.07</td>
</tr>
<tr>
<td>0.29</td>
<td>0.08</td>
</tr>
<tr>
<td>0.81</td>
<td>0.09</td>
</tr>
<tr>
<td>1.62</td>
<td>0.1</td>
</tr>
<tr>
<td>23.3</td>
<td>0.1</td>
</tr>
</tbody>
</table>

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.

\[
r = \lim_{\rho_{av} \to 0} \frac{0.1e^{\rho_{av}} - (0.03e^{-\rho_{av}})}{e^{\rho_{av}} + e^{-\rho_{av}}} = 0.03
\]

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

\[
r = \lim_{\rho_{av} \to \infty} \frac{0.1e^{\rho_{av}} - (0.03e^{-\rho_{av}})}{e^{\rho_{av}} + e^{-\rho_{av}}} = 0.1
\]

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 \(K\)

\[
f(\rho_{av}) = \tanh(\rho_{av}) = \frac{e^{K\rho_{av}} - 1}{e^{K\rho_{av}} + 1}
\]

At each iteration the curvature radius is calculated to obtain the radius of the sphere suitable for tracing the trajectory.
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.

\[
\text{sign}(\Delta \lambda_{j+1} \Delta \lambda_j)
\]  

(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.
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.
Figure 3.13: Approximate solution
Chapter 3. Proposed methodology for the variable radius in the spherical method
Bibliography


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].

![Two tunnel diode circuit](image)

Figure 4.1: Two tunnel diode circuit.

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

\[
\begin{align*}
g_1(v_1) &= 2.5v_1^3 - 10.5v_1^2 + 11.8v_1, \\
g_2(v_2) &= 0.43v_2^3 - 2.69v_2^2 + 4.56v_2.
\end{align*}
\]

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

\[
\begin{align*}
F_1(v_1, v_2) &= E - Rg_1(v_1) - (v_1 + v_2) = 0, \\
F_2(v_1, v_2) &= g_1(v_1) - g_2(v_2) = 0.
\end{align*}
\]

(4.2)

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

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

(4.3)

The Spherical method is applied to the system of equations and the results are in table
4.1. Non-linear circuit with two tunnel diodes

4.1, where reversion is detected using the calculus of normal vector. Then the strategy to
avoid reversion problem is applied, obtaining the following results. Table 4.2 The Figure

<table>
<thead>
<tr>
<th>Iter.</th>
<th>$x_1$</th>
<th>$x_2$</th>
<th>$\lambda$</th>
<th>$\text{angle}(x_1)$</th>
<th>$\text{angle}(x_2)$</th>
<th>$\text{angle}(\lambda)$</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>7.168768945</td>
<td>21.584876168</td>
<td>0.5</td>
<td>102.5/102.5</td>
<td>24.1/24.1</td>
<td>110.3/110.3</td>
</tr>
<tr>
<td>2</td>
<td>7.158354005</td>
<td>21.557119274</td>
<td>0.489587336</td>
<td>103.2/77.4</td>
<td>15.9/155.8</td>
<td>98.6/69.6</td>
</tr>
<tr>
<td>3</td>
<td>7.15147398135986</td>
<td>21.528269750</td>
<td>0.486009949</td>
<td>103.3/76.7</td>
<td>14.9/164.0</td>
<td>96.6/81.3</td>
</tr>
</tbody>
</table>

Table 4.1: 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)

<table>
<thead>
<tr>
<th>Solution</th>
<th>Iteration</th>
<th>$v_1$</th>
<th>$v_2$</th>
<th>Error $= \sqrt{f_1^2 + f_2^2}$</th>
</tr>
</thead>
<tbody>
<tr>
<td>$S_1$</td>
<td>106</td>
<td>2.305222063</td>
<td>0.705560377</td>
<td>1.3E-10</td>
</tr>
<tr>
<td>$S_2$</td>
<td>143</td>
<td>2.277597006</td>
<td>1.857491731</td>
<td>3.3E-10</td>
</tr>
<tr>
<td>$S_3$</td>
<td>205</td>
<td>2.224729753</td>
<td>3.693943974</td>
<td>6.5E-10</td>
</tr>
<tr>
<td>$S_4$</td>
<td>249</td>
<td>1.775593561</td>
<td>3.707177714</td>
<td>8.8E-11</td>
</tr>
<tr>
<td>$S_5$</td>
<td>316</td>
<td>1.702657758</td>
<td>1.809029946</td>
<td>1.4E-10</td>
</tr>
<tr>
<td>$S_6$</td>
<td>349</td>
<td>1.666377840</td>
<td>0.739343469</td>
<td>5.8E-11</td>
</tr>
<tr>
<td>$S_7$</td>
<td>553</td>
<td>0.228250851</td>
<td>0.828262137</td>
<td>1.2E-11</td>
</tr>
<tr>
<td>$S_8$</td>
<td>558</td>
<td>0.219854573</td>
<td>1.672981409</td>
<td>1.03E-10</td>
</tr>
<tr>
<td>$S_9$</td>
<td>628</td>
<td>0.199709392</td>
<td>3.754217099</td>
<td>1.10E-11</td>
</tr>
</tbody>
</table>

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 $\lambda$ and the Figure 4.4 show the variable $v_2$ versus $\lambda$ for the nine solutions.
4.1. Non-linear circuit with two tunnel diodes

Figure 4.3: The projection $v_1$ over $\lambda$. 

(a) Homotopy path projected over $v_1-\lambda$

(b) Zoom to solutions of (a)

(c) Zoom to solutions of (a)

(d) Zoom to solutions of (a)
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.

\[
H_1(f_1, \lambda) = 30 - 33.25v_1^3 + 139.65v_1^2 - 157.94v_1 - v_2 - (1 - y)f(x_0),
\]

\[
H_2(f_2, \lambda) = 2.5v_1^3 - 10.5v_2^2 + 11.8v_1 - 0.43v_2^3 + 2.69v_2^2 - 4.56v_2 - (1 - y)f(x_0),
\]

\[
(v_1 - c_1)^2 + (v_2 - c_2)^2 + (\lambda - c_3)^2 - r = 0. \tag{4.4}
\]

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
4.1. Non-linear circuit with two tunnel diodes

Table 4.4: Numerical results for the first case study

<table>
<thead>
<tr>
<th></th>
<th>CPU time</th>
<th>Iteration number</th>
</tr>
</thead>
<tbody>
<tr>
<td>same size radio (0.06)</td>
<td>7.03</td>
<td>205</td>
</tr>
<tr>
<td>different size radio (0.06-0.1)</td>
<td>6.1</td>
<td>146</td>
</tr>
</tbody>
</table>

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 reduced and therefore there are a difference in CPU time; using in both cases the Newton homotopy formulation.

![Figure 4.5: (A) Radius size constant. (B) Radius size variable.](image-url)
(a) Homotopy path projected over $v_1 - \lambda$ 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 $\lambda$ using Newton Homotopy.
4.1. Non-linear circuit with two tunnel diodes

(a) Homotopy path projected over $v_1-\lambda$ 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 $\lambda$ using Newton Homotopy.
4.1.1 Mathematical example

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

\[
F_1(x_1, x_2) = 4(x_1^2 + x_2^2 - 1)x_1 + 16((2x_1^2 - 1)^2 + (2x_2^2 - 1)^2 - 2/3)(2x_1^2 - 1)x_1 = 0,
\]
\[
F_2(x_1, x_2) = 4(x_1^2 + x_2^2 - 1)x_2 + 16((2x_1^2 - 1)^2 + (2x_2^2 - 1)^2 - 2/3)(2x_2^2 - 1)x_2 = 0. \tag{4.5}
\]

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

\[
H_1(f_1, \lambda) = \lambda(\lambda - 1) + \exp(\lambda(\lambda - 1)) \ln(0.09((4v_1^2 + 4v_2^2 - 4)v_1
+ 16(2v_1^2 - 1)^2 + 16(2v_2^2 - 1)^2 - 32/3)(2v_1^2 - 1)v_1)^2 + 1) = 0,
\]
\[
H_2(f_2, \lambda) = \lambda(\lambda - 1) + \exp(\lambda(\lambda - 1)) \ln(0.09((4v_1^2 + 4v_2^2 - 4)v_2
+ 16(2v_1^2 - 1)^2 + 16(2v_2^2 - 1)^2 - 32/3)(2v_2^2 - 1)v_2)^2 + 1) = 0,
\]
\[
(v_1 - c_1)^2 + (v_2 - c_2)^2 + (\lambda - c_3)^2 = 0.0009. \tag{4.6}
\]

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 \(\lambda\) go back to \(\lambda = 0.47\).

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

<table>
<thead>
<tr>
<th>Iter.</th>
<th>(x_1)</th>
<th>(x_2)</th>
<th>(\lambda)</th>
<th>(\text{angle}(x_1))</th>
<th>(\text{angle}(x_2))</th>
<th>(\text{angle}(\lambda))</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>0.931862833</td>
<td>-0.931862833</td>
<td>0.5</td>
<td>90.2/90.2</td>
<td>89.7/89.7</td>
<td>179.5/179.5</td>
</tr>
<tr>
<td>2</td>
<td>0.931707573</td>
<td>-0.931707573</td>
<td>0.4700000000</td>
<td>89.7/90.8</td>
<td>90.2/89.1</td>
<td>0.41/178</td>
</tr>
<tr>
<td>3</td>
<td>0.931242105</td>
<td>-0.931242105</td>
<td>0.440004014</td>
<td>89.1/91.4</td>
<td>90.8/88.6</td>
<td>1.25/177.9</td>
</tr>
</tbody>
</table>

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:
4.1. Non-linear circuit with two tunnel diodes

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 \( \lambda \) 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
Chapter 4. Case studies

<table>
<thead>
<tr>
<th>Solution</th>
<th>Iteration</th>
<th>$v_1$</th>
<th>$v_2$</th>
<th>Error $= \sqrt{f_1^2 + f_2^2}$</th>
</tr>
</thead>
<tbody>
<tr>
<td>S1</td>
<td>19</td>
<td>0.853356743</td>
<td>-.853356743</td>
<td>3E-15</td>
</tr>
<tr>
<td>S2</td>
<td>26</td>
<td>0.707106781</td>
<td>-.707106781</td>
<td>8.61E-14</td>
</tr>
<tr>
<td>S3</td>
<td>34</td>
<td>0.521327409</td>
<td>-.521327409</td>
<td>3.29E-11</td>
</tr>
<tr>
<td>S4</td>
<td>61</td>
<td>-1.71E-16</td>
<td>1.70E-16</td>
<td>2.4E-11</td>
</tr>
<tr>
<td>S5</td>
<td>87</td>
<td>-0.521327409</td>
<td>0.521327409</td>
<td>2E-15</td>
</tr>
<tr>
<td>S6</td>
<td>96</td>
<td>-0.707106785</td>
<td>0.707106785</td>
<td>2E-15</td>
</tr>
<tr>
<td>S7</td>
<td>103</td>
<td>-0.853356744</td>
<td>0.853356744</td>
<td>9.6E-11</td>
</tr>
</tbody>
</table>

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

<table>
<thead>
<tr>
<th>Solution</th>
<th>Iteration</th>
<th>$x_1$</th>
<th>$x_2$</th>
</tr>
</thead>
<tbody>
<tr>
<td>S1</td>
<td>27</td>
<td>-0.457846075</td>
<td>-0.88583237</td>
</tr>
<tr>
<td>S2</td>
<td>36</td>
<td>-0.52105087</td>
<td>-0.52233449</td>
</tr>
<tr>
<td>S3</td>
<td>42</td>
<td>-0.706918715</td>
<td>-0.705518887</td>
</tr>
<tr>
<td>S4</td>
<td>46</td>
<td>-0.856403500</td>
<td>-0.840392018</td>
</tr>
<tr>
<td>S5</td>
<td>52</td>
<td>-0.887998886</td>
<td>-0.459719460</td>
</tr>
<tr>
<td>S6</td>
<td>60</td>
<td>-0.788768474</td>
<td>0.000686285</td>
</tr>
<tr>
<td>S7</td>
<td>67</td>
<td>-0.887951404</td>
<td>0.459867289</td>
</tr>
<tr>
<td>S8</td>
<td>74</td>
<td>-0.853645422</td>
<td>0.851469150</td>
</tr>
<tr>
<td>S9</td>
<td>78</td>
<td>-0.706365080</td>
<td>0.703951513</td>
</tr>
<tr>
<td>S10</td>
<td>83</td>
<td>-0.521087102</td>
<td>0.522784769</td>
</tr>
<tr>
<td>S11</td>
<td>90</td>
<td>-0.459290981</td>
<td>0.887511352</td>
</tr>
<tr>
<td>S12</td>
<td>107</td>
<td>0.000467550</td>
<td>0.783519531</td>
</tr>
<tr>
<td>S13</td>
<td>117</td>
<td>-0.000003365</td>
<td>-0.000071349</td>
</tr>
<tr>
<td>S14</td>
<td>126</td>
<td>-0.000102981</td>
<td>-0.785701888</td>
</tr>
<tr>
<td>S15</td>
<td>143</td>
<td>0.457897013</td>
<td>-0.885735694</td>
</tr>
<tr>
<td>S16</td>
<td>150</td>
<td>0.521107681</td>
<td>-0.52255104</td>
</tr>
<tr>
<td>S17</td>
<td>156</td>
<td>0.706976322</td>
<td>-0.70475460</td>
</tr>
<tr>
<td>S18</td>
<td>160</td>
<td>0.854672950</td>
<td>-0.845544717</td>
</tr>
<tr>
<td>S19</td>
<td>166</td>
<td>0.888023651</td>
<td>-0.459737787</td>
</tr>
<tr>
<td>S20</td>
<td>174</td>
<td>0.788560647</td>
<td>0.001018800</td>
</tr>
<tr>
<td>S21</td>
<td>180</td>
<td>0.887897377</td>
<td>0.459769872</td>
</tr>
<tr>
<td>S22</td>
<td>187</td>
<td>0.854728259</td>
<td>0.847735395</td>
</tr>
<tr>
<td>S23</td>
<td>191</td>
<td>0.706625443</td>
<td>0.705505264</td>
</tr>
<tr>
<td>S24</td>
<td>195</td>
<td>0.521211842</td>
<td>0.521668777</td>
</tr>
<tr>
<td>S25</td>
<td>204</td>
<td>0.457859208</td>
<td>0.885858576</td>
</tr>
</tbody>
</table>

Table 4.8: Numerical solutions to the equation (4.6)
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

<table>
<thead>
<tr>
<th></th>
<th>CPU time</th>
<th>Iteration number</th>
</tr>
</thead>
<tbody>
<tr>
<td>fixed radius (0.03)</td>
<td>7.4</td>
<td>205</td>
</tr>
<tr>
<td>variable radius (0.03-0.1)</td>
<td>14.3</td>
<td>462</td>
</tr>
</tbody>
</table>
4.1.2 Circuit with two tunnel exponential diodes

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

\[ i_{1,2} = I_p \left( \frac{V}{V_p} \right) e^{1 - \frac{V}{V_p}} + I_0 e^{\frac{q}{kT} V} \]  

(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 \( \lambda \). The Figure 4.13 show that the method has found 5 five solutions into the homotopy path.
4.1. Non-linear circuit with two tunnel diodes

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

<table>
<thead>
<tr>
<th>Solution</th>
<th>Iteration</th>
<th>$I_{v1}$</th>
<th>$v_2$</th>
<th>$v_3$</th>
<th>Error = $\sqrt{f_1^2 + f_2^2}$</th>
</tr>
</thead>
<tbody>
<tr>
<td>$S_1$</td>
<td>45</td>
<td>-0.048873941</td>
<td>0.022521177</td>
<td>0.011260588</td>
<td>1.1E-13</td>
</tr>
<tr>
<td>$S_2$</td>
<td>52</td>
<td>-0.042175710</td>
<td>0.156485795</td>
<td>0.147132103</td>
<td>6.5E-10</td>
</tr>
<tr>
<td>$S_3$</td>
<td>70</td>
<td>-0.018919917</td>
<td>0.021601653</td>
<td>0.418182617</td>
<td>1.5E-11</td>
</tr>
<tr>
<td>$S_4$</td>
<td>77</td>
<td>-0.028280221</td>
<td>0.434395575</td>
<td>0.428548398</td>
<td>2.4E-11</td>
</tr>
<tr>
<td>$S_5$</td>
<td>80</td>
<td>-0.009914377</td>
<td>0.801712456</td>
<td>0.400856225</td>
<td>2.3E-09</td>
</tr>
</tbody>
</table>

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 $\lambda$
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

<table>
<thead>
<tr>
<th>Solution</th>
<th>Iteration</th>
<th>$I_{v_1}$</th>
<th>$v_2$</th>
<th>$v_3$</th>
</tr>
</thead>
<tbody>
<tr>
<td>$S_1$</td>
<td>32</td>
<td>-0.028280220</td>
<td>0.434395581</td>
<td>0.005847176</td>
</tr>
<tr>
<td>$S_2$</td>
<td>36</td>
<td>-0.042175710</td>
<td>0.156485796</td>
<td>0.009353691</td>
</tr>
<tr>
<td>$S_3$</td>
<td>39</td>
<td>-0.048873941</td>
<td>0.022521177</td>
<td>0.011260588</td>
</tr>
<tr>
<td>$S_4$</td>
<td>47</td>
<td>-0.042175710</td>
<td>0.156485796</td>
<td>0.147132104</td>
</tr>
<tr>
<td>$S_5$</td>
<td>52</td>
<td>-0.033613653</td>
<td>0.327726928</td>
<td>0.163863464</td>
</tr>
<tr>
<td>$S_6$</td>
<td>55</td>
<td>-0.018919917</td>
<td>0.621601656</td>
<td>0.203419036</td>
</tr>
<tr>
<td>$S_7$</td>
<td>84</td>
<td>-0.009914377</td>
<td>0.801712451</td>
<td>0.400856225</td>
</tr>
<tr>
<td>$S_8$</td>
<td>87</td>
<td>-0.018919917</td>
<td>0.621601653</td>
<td>0.418182617</td>
</tr>
<tr>
<td>$S_9$</td>
<td>92</td>
<td>-0.028280221</td>
<td>0.434395574</td>
<td>0.428548398</td>
</tr>
</tbody>
</table>

Table 4.11: Numerical solutions to the equation (4.8)

methodology of variable Table 4.12, the results show fewer iterations and consequently
4.1. Non-linear circuit with two tunnel diodes

Figure 4.13: The projection $I_{v_1}, v_2, v_3$ over $\lambda$.

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.

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

Table 4.12: Solutions third case study
4.1.3 Four-transistor multistate circuit

The next circuit contain four-transistor shown Figure 4.14 Bipolar transistor is replaced 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 $I_0 = 10E - 06$.

\begin{align*}
    f_1 &= 6.103168I_0(\exp(40v_1) - 1) + 4.36634v_2 + 2.863168(\exp(40v_2) - 1) - 12 \\
    f_2 &= 5.4v_1 + 3.58I_0(\exp(40v_1) - 1) + 6.62I_0(\exp(40v_2) - 1) + v_3 \\
        &\quad + 0.7I_0(\exp(40v_3) - 1) + 0.5I_0(\exp(40v_4) - 1) - 22 \\
    f_3 &= 6.103168I_0(\exp(40v_3) - 1) + 2.863168I_0(\exp(40v_4) - 1) + 4.36634v_4 - 12 \\
    f_4 &= 1v_1 + 0.7I_0(\exp(40v_1) - 1) + 0.5I_0(\exp(40v_2) - 1) + 5, 4v_3 \\
        &\quad + 3.58I_0(\exp(40v_3) - 1) + 6.62I_0(\exp(40v_4) - 1) - 20 \quad (4.9)
\end{align*}
4.1. Non-linear circuit with two tunnel diodes

Applying homotopy function is obtained:

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

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

<table>
<thead>
<tr>
<th>No.</th>
<th>Initial point</th>
<th>(0.03)</th>
<th>(0.03-0.1)</th>
<th>Solution</th>
</tr>
</thead>
<tbody>
<tr>
<td>S1</td>
<td>(-0.5,0,0,-0.5)</td>
<td>138</td>
<td>66</td>
<td>(-1.42801288,0.18413961,0.19813414,-4.456385520)</td>
</tr>
<tr>
<td>S2</td>
<td>(-0.5,0.5,0.5,0.5)</td>
<td>204</td>
<td>80</td>
<td>(0.19011567,-4.201978519,0.186391138,-3.239254017)</td>
</tr>
<tr>
<td>S3</td>
<td>(0.5,0.5,0.5,0.5)</td>
<td>42</td>
<td>18</td>
<td>(0.13548515,0.178280202,0.14379723,0.16926345)</td>
</tr>
<tr>
<td>S4</td>
<td>(-0.1,0.1)</td>
<td>35</td>
<td>15</td>
<td>(-3.84596284,0.24413802,-1.39008029,0.18413802)</td>
</tr>
<tr>
<td>S5</td>
<td>(0.0,-1.0)</td>
<td>183</td>
<td>73</td>
<td>(0.19419199,-5.41908207,0.18778370,0.18413976)</td>
</tr>
<tr>
<td>S6</td>
<td>(0.1,-0.1)</td>
<td>134</td>
<td>55</td>
<td>(0.14428324,0.17920052,0.18921466,-3.9577056)</td>
</tr>
<tr>
<td>S7</td>
<td>(-0.1,0.1,0.1)</td>
<td>160</td>
<td>65</td>
<td>(0.19204522,-0.19204522,0.15082057,0.16393479)</td>
</tr>
</tbody>
</table>

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.

Bipolar transistor is replaced by Ebers-Moll model Figure 4.17.

The Ebers-Moll model is defined by

$$\begin{bmatrix} i_e \\ i_c \end{bmatrix} = \begin{bmatrix} 1 & -0.01 \\ -0.99 & 1 \end{bmatrix} \begin{bmatrix} 10^{-9} (e^{40v_{bc}} - 1) \\ 10^{-9} (e^{40v_{bc}} - 1) \end{bmatrix}$$
Chapter 4. Case studies

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

\[
\begin{align*}
    f_1 &= \frac{37}{20000} v_1 - \frac{1}{4000} v_2 - \frac{1}{4000} v_6 - \frac{1}{1000} v_9 - \frac{1}{4000} v_{12} - \frac{1}{10000} v_{13} + I_E \\
    f_2 &= -\frac{1}{4000} v_1 + \frac{3}{8000} v_2 + \frac{3}{2000} v_5 + 9.90E - 09 \exp{40v_4 - 40v_3} + 1E - 10 - 1E - 08 \exp{40v_4 - 40v_2} \\
    f_3 &= \frac{1}{1000} v_3 - 1E - 08 \exp{40v_4 - 40v_3} + 9.90E - 09 + 1E - 10 \exp{40v_4 - 40v_2} \\
    f_4 &= \frac{1}{8000} v_4 - \frac{1}{8000} v_6 + 1E - 10 \exp{40v_4 - 40v_3} - 1E - 08 + 9.90E - 09 \exp{40v_4 - 40v_2} \\
    f_5 &= \frac{1}{1000} v_5 - \frac{1}{8000} v_6 + 1E - 10 \exp{40v_5 - 40v_7} - 1E - 08 + 9.90E - 09 \exp{40v_5 - 40v_6} \\
    f_6 &= \frac{1}{1000} v_6 - \frac{1}{8000} v_4 + \frac{3}{8000} v_6 + 9.90E - 09 \exp{40v_5 - 40v_7} + 1.010E - 08 \exp{40v_5 - 40v_6} - 1E - 08 \exp{40v_5 - 40v_6} \\
    f_7 &= \frac{1}{1000} v_7 - 1E - 08 \exp{40v_5 - 40v_7} + 9.90E - 09 + 1E - 10 \exp{40v_5 - 40v_6} \\
    f_8 &= \frac{1}{30000} v_8 - \frac{1}{30000} v_9 + 1E - 08 \exp{40v_8 - 40v_9} - 1E - 08 \\
    f_9 &= \frac{1}{1000} v_9 - \frac{1}{30000} v_8 + \frac{31}{30000} v_9 + 9.90E - 09 \exp{40v_{11} - 40v_{10}} + 1E - 10 + 1E - 08 \exp{40v_{11} - 40v_9} \\
    f_{10} &= \frac{1}{1000} v_{10} - 1E - 08 \exp{40v_{11} - 40v_{10}} + 9.90E - 09 + 1E - 10 \exp{40v_{11} - 40v_9} \\
    f_{11} &= \frac{1}{10000} v_{11} - \frac{1}{10000} v_{12} + 1E - 10 \exp{40v_{11} - 40v_{10}} - 1E - 08 + 9.9E - 09 \exp{40v_{11} - 40v_9} \\
    f_{12} &= \frac{1}{40000} v_{12} - \frac{1}{10000} v_{11} + \frac{7}{20000} v_{12} + 9.9E - 12 \exp{40v_{13} - 40v_{12}} + 1E - 10 - 1E - 08 \exp{40v_{13} - 40v_{12}} \\
    f_{13} &= \frac{1}{10000} + \frac{1}{10000} v_{13} + 1E - 10 \exp{40v_{13}} - 1E - 08 + 9.9E - 09 \exp{40v_{13} - 40v_{12}} \\
    f_{14} &= v_1 - 12,
\end{align*}
\]

(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 \( \lambda \) parameter can be plotted, Figure 4.18 shows \( I_E \) versus \( \lambda \) and \( v_{13} \) versus \( \lambda \).
Table 4.14: Solutions for example with 14 variables

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

Table 4.15: Solutions fixed radio-variable radio

<table>
<thead>
<tr>
<th>Initial point</th>
<th>Sol.</th>
<th>Iter. ($r=0.03$)</th>
<th>CPU time (sec)</th>
<th>Iter. ($r=0.03-0.1$)</th>
<th>CPU time (sec)</th>
</tr>
</thead>
<tbody>
<tr>
<td>(0, 10, 10, 10, 10, 0, 10, 10, 10, 10, 10, 0, 10)</td>
<td>$S_1$</td>
<td>478</td>
<td>696.2</td>
<td>180</td>
<td>601.7</td>
</tr>
<tr>
<td>(0, 12, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0)</td>
<td>$S_2$</td>
<td>478</td>
<td>696.2</td>
<td>172</td>
<td>601.7</td>
</tr>
<tr>
<td>(0.5, 0.5, 0.5, 0.5, 0.5, 0.5, 0.5, 0.5, 0.5, 0.5, 0.5, 0.5)</td>
<td>$S_3$</td>
<td>476</td>
<td>696.2</td>
<td>171</td>
<td>601.7</td>
</tr>
</tbody>
</table>
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

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


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.
4.1. Non-linear circuit with two tunnel diodes

Table 4.17: Solutions Multistate Circuit

<table>
<thead>
<tr>
<th>Sol</th>
<th>( S_1 )</th>
</tr>
</thead>
<tbody>
<tr>
<td>( I_E )</td>
<td>-0.0347</td>
</tr>
<tr>
<td>( v_1 )</td>
<td>12</td>
</tr>
<tr>
<td>( v_2 )</td>
<td>8.2950</td>
</tr>
<tr>
<td>( v_3 )</td>
<td>8.0239</td>
</tr>
<tr>
<td>( v_4 )</td>
<td>0.2842</td>
</tr>
<tr>
<td>( v_5 )</td>
<td>0.3276</td>
</tr>
<tr>
<td>( v_6 )</td>
<td>0.5148</td>
</tr>
<tr>
<td>( v_7 )</td>
<td>0.5148</td>
</tr>
<tr>
<td>( v_8 )</td>
<td>0.7824</td>
</tr>
<tr>
<td>( v_9 )</td>
<td>1.0244</td>
</tr>
<tr>
<td>( v_{10} )</td>
<td>5.1209</td>
</tr>
<tr>
<td>( v_{11} )</td>
<td>0.6756</td>
</tr>
<tr>
<td>( v_{12} )</td>
<td>0.9813</td>
</tr>
<tr>
<td>( v_{13} )</td>
<td>0.7824</td>
</tr>
<tr>
<td>( v_{14} )</td>
<td>1.0244</td>
</tr>
<tr>
<td>( v_{15} )</td>
<td>5.1209</td>
</tr>
<tr>
<td>( v_{16} )</td>
<td>0.6756</td>
</tr>
<tr>
<td>( v_{17} )</td>
<td>0.9813</td>
</tr>
</tbody>
</table>
4.2 Results and Discussion

The Spherical algorithm has been used with modifications to avoid 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 successfully 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.
4.3. Summary

(a) Homotopy path projected over $I_{v_1}-\lambda$

(b) Homotopy path projected over $v_1-\lambda$

(c) Homotopy path projected over $v_2-\lambda$

(d) Homotopy path projected over $v_3-\lambda$

(e) Homotopy path projected over $v_4-\lambda$

(f) Homotopy path projected over $v_5-\lambda$

Figure 4.20: The projection $I_{v_1}, v_1, v_2, v_3, v_4, v_5$ over $\lambda$. 

77
Figure 4.21: The projection $v_6, v_7, v_8, v_9, v_{10}, v_{11}$ over $\lambda$. 
4.3. Summary

(a) Homotopy path projected over $v_{12}$-$\lambda$

(b) Homotopy path projected over $v_{13}$-$\lambda$

(c) Homotopy path projected over $v_{14}$-$\lambda$

(d) Homotopy path projected over $v_{15}$-$\lambda$

(e) Homotopy path projected over $v_{16}$-$\lambda$

(f) Homotopy path projected over $v_{17}$-$\lambda$

Figure 4.22: The projection $v_{12}$, $v_{13}$, $v_{14}$, $v_{15}$, $v_{16}$, $v_{17}$ over $\lambda$. 
Chapter 4. Case studies
Bibliography


Chapter 5

Conclusions

Spherical algorithm has 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.