

Homotopy using Matlab (Analysis and Implementation)

Sudhir Kumar

Indian Institute of Technology, Bombay, India

13D070036@iitb.ac.in

Guide: Ljiljana Trajkovic

School of Engineering Science

Simon Fraser Institute, Burnaby

British Columbia, Canada

July 14, 2016

1 Homotopy

1.1 Introduction

Homotopy methods also known as continuation or embedding methods, solve difficult problem first by solving easy problem and then design the problem in such a way that continuing the path along some parameter we get the solution of hard problem.

To make it more clear we will give the full description of standard homotopy below:

Homotopy methods[1] are used to solve systems of non-linear algebraic equations and may be applied to a large variety of problems. We are most interested in solving the zero finding problem

$$F(x) = 0 \tag{1}$$

where $x \in \mathbb{R}^n$, $F : \mathbb{R}^n \rightarrow \mathbb{R}^n$. Note that the fixed point problem $F(x) = x$ may be easily reformulated as a zero finding problem

$$F(x) - x = 0 \tag{2}$$

A homotopy function $H(x, \lambda)$ is created by embedding a parameter λ into $F(x)$ to obtain an equation of higher dimension

$$H(x, \lambda) = 0$$

where $\lambda \in \mathbb{R}$, $H: \mathbb{R}^n \times \mathbb{R} \rightarrow \mathbb{R}^n$. For $\lambda = 0$,

$$H(x, 0) = 0 \tag{3}$$

is an easy equation to solve. For $\lambda = 1$,

$$H(x, 1) = 0 \tag{4}$$

is the original problem (1). The parameter λ is called the continuation or homotopy parameter.

An example of a homotopy function is

$$H(x, \lambda) = (1 - \lambda)G(x) + \lambda F(x) \tag{5}$$

Hence,

$$H(x, 0) := G(x) = 0 \tag{6}$$

has an easy solution while

$$H(x, 1) := F(x) = 0 \tag{7}$$

is the original problem. By following solutions of

$$H(x, \lambda) = 0 \tag{8}$$

as λ varies from 0 to 1, the solution to $F(x) = 0$ is reached.

The solutions (8) trace a path known as the zero curve. Various numerical situations may occur depending on the behaviour of this curve. One problem occurs if the curve folds back. At the turning point, the values of λ decrease as the path progresses. Increasing λ from 0 to 1 results in “losing” the curve. The difficulty is resolved by making λ a function of a new parameter: the arc length s . This method is known as the *arc length* continuation.

1.2 Convergent ball concept for “globally convergent with probability one.”

Let B be the closed unit ball in n -dimensional real Euclidean space E^n and let $f: B \rightarrow B$ be a C^2 map. Define $H(x, \lambda): [0,1] \times B \rightarrow E^n$ by

$$H(\lambda, x) = \lambda(x - f(x)) + (1 - \lambda)(x - a) \tag{9}$$

The fundamental result [2] is that for almost all a in the interior of B , there is a zero curve $\gamma \subset [0,1] \times B$ of H , along which the Jacobian matrix has rank n , emanating from $(0, a)$ and reaching a point $(1, x^{\sim})$, where x^{\sim} is a fixed point of f . thus with probability one, picking a starting point $a \in \text{int } B$ and following γ leads to a fixed point x^{\sim} of f .

2 Types of Globally convergent homotopy functions

In the following homotopies, G_{leak} has been used frequently. The G_{leak} term used scales each linear term. $G_{\text{leak}} \in R^n \times R^n$ is a known matrix containing only one non-zero entry per row and column.

For simplicity, $G_{\text{leak}}[3]$ is chosen as the coefficient of the corresponding x_i terms in $F(x)$. In some cases, the coefficients were too small, causing the Jacobian to lose rank, so for small coefficients ($<10^{-8}$), the G_{leak} term is set to 1.

2.1 Fixed point homotopy:

$$H(x, \lambda) = (1 - \lambda)(x - a)G_{\text{leak}} + \lambda F(x) \quad (10)$$

In this homotopy, the whole equation is multiplied by λ . This is only a slight variation of standard homotopy.

2.2 Variable stimulus homotopy:

$$H(x, \lambda) = (1 - \lambda)(x - a)G_{\text{leak}} + F(x, \lambda) \quad (11)$$

In this homotopy, only non-linear terms of $F(x)$ are multiplied by λ . The advantage of this homotopy is that all non-linear terms are removed in the analysis of $\lambda=0$ making the solution dependent only on linear circuit.

2.3 Modified variable stimulus homotopy:

$$H(x, \lambda) = (1 - \lambda)(x - a)G_{\text{leak}} + F(\lambda x) \quad (12)$$

In this homotopy, all the nodes are multiplied by λ . This homotopy was implemented by me. And the results and further discussions will be in the various sections below.

2.4 Variable gain homotopy:

$$H(x, \lambda) = (1 - \lambda)(x - a)G_{\text{leak}} + F(x, \lambda\alpha) \quad (13)$$

In this homotopy, the transistors forward and reverse gains are correspondingly multiplied by λ . This is the fastest converging one for bipolar circuits.

3 Testing of 2 transistor circuits on fixed point homotopy:

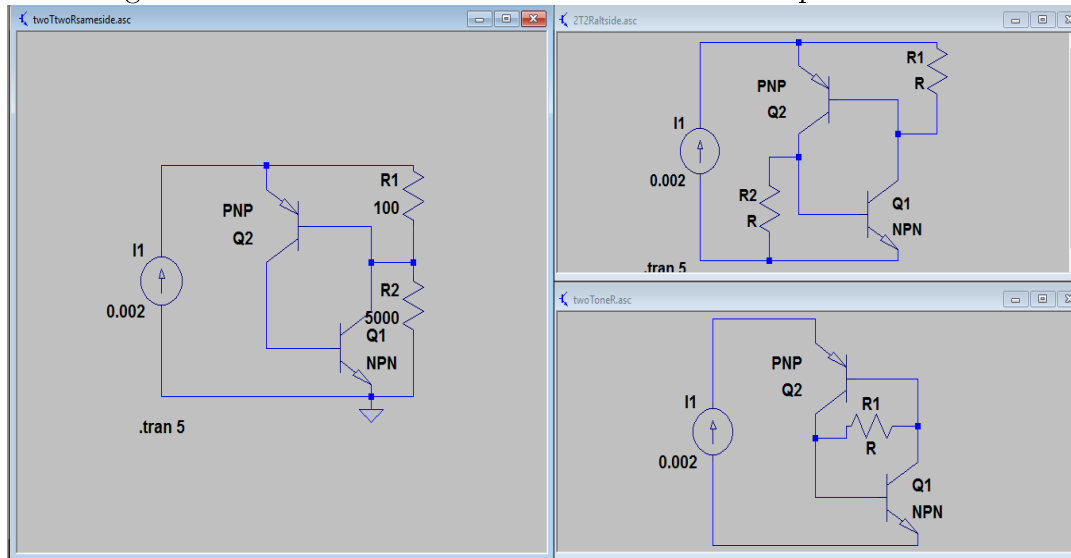
The fixed point homotopy written by Heath Hofmann in Matlab [4], [5] was needed to be tested on more circuits. The fixed point homotopy has

the requirement of circuit equations and Jacobians, both are generated by the Parser.

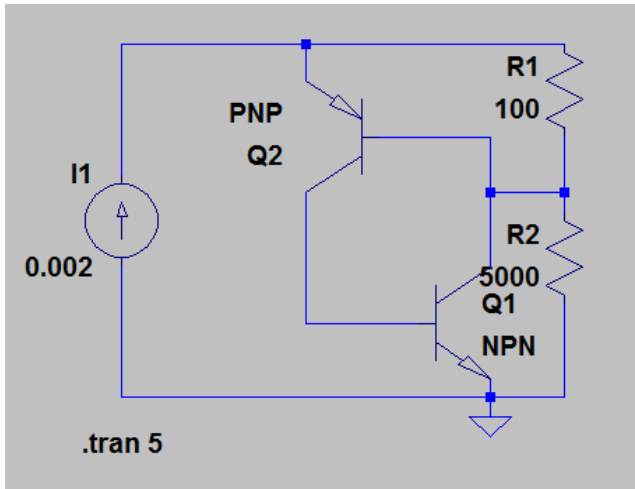
We decided to analyze the circuits[6] which have been point of interest for the scientists and engineers for a long time due to its simulation results on the present circuit simulators. These circuits are designed using only two transistors and claimed as voltage latch up circuits. Later after research in NDR theory[7] it was found that some resistors are also needed to show the NDR behavior. According to simulations these circuits go upto 10^6 V with an input of just few mA which is never possible. So we choose these circuits to test our homotopy and choose the initial values that had been the solutions on Ltspice.

3.1 Circuits diagram:

Following are the circuits tested which showed some exceptional results.



3.2 Homotopy output:

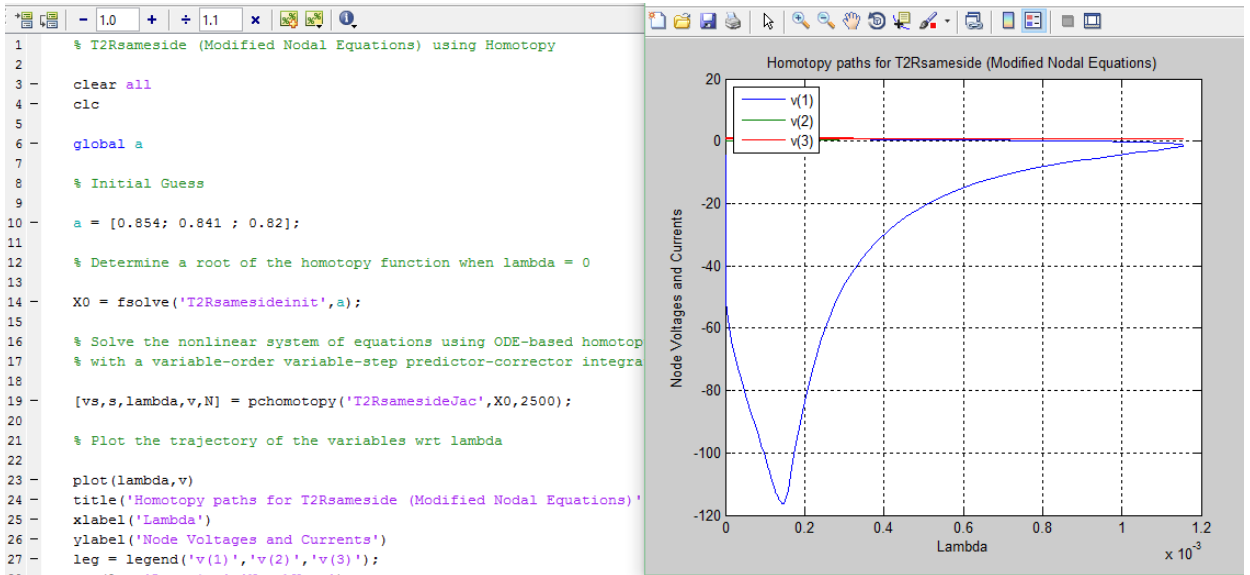


Testing parameters:

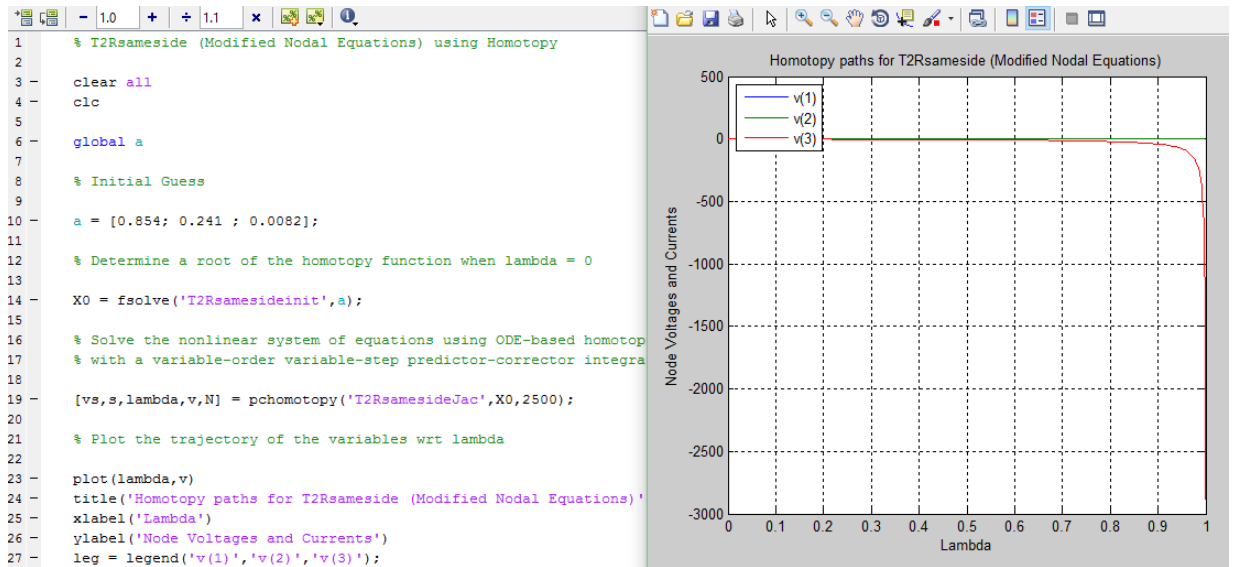
% BJT Parameters

q1IS = 1e-14;
 q1BF = 200;
 q1BR = 3;
 q1N = 38.78;
 q2IS = 1e-14;
 q2BF = 250;
 q2BR = 3;
 q2N = 38.78;

1. Initial condition: $v_1=0.854$; $v_2=0.841$, $v_3=0.82$, $I_1=2e-03$



2. Initial condition: $v_1=0.854$; $v_2=0.241$, $v_3=0.0082$, $I_1=2e-03$



As you can see the output of homotopy is very sensitive to the initial conditions i.e “a”. After a lot of simulations tweaking “a” every time the value of lambda reached changed a lot. Initially lambda was able to reach only 10^{-3} to 10^{-4} range. But later lambda reached 0.001156 and then 0.9987, very close to 1 but couldn’t cross 1 and hence no solutions.

4 Results

Some more simulation results have been summarized in the following table that has been obtained by changing various parameters and the value of lambda that could be reached.

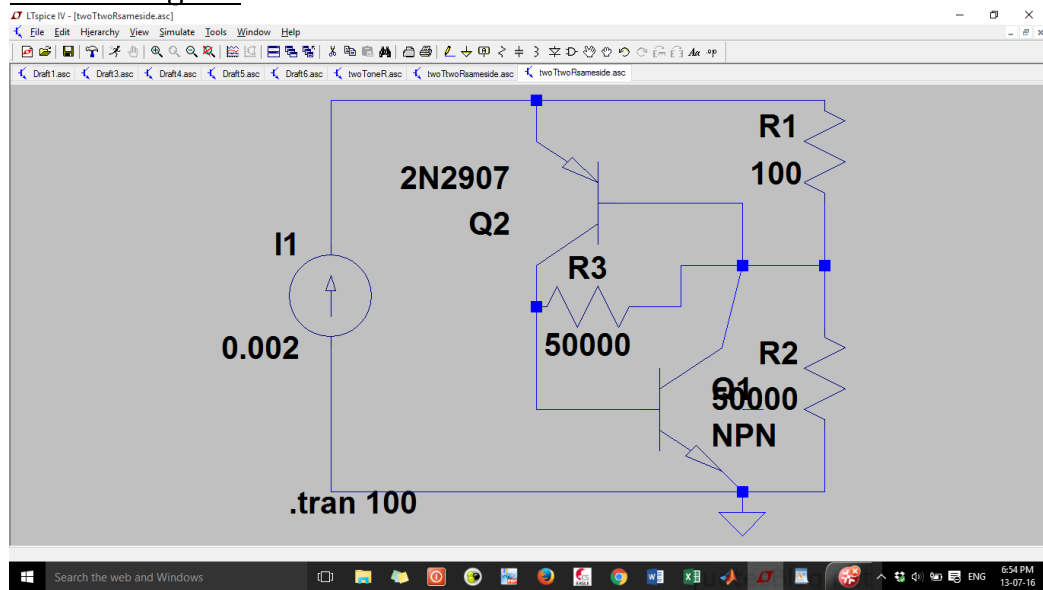
r1	r2	Input (I)	IS	q2 Bf	q2 Br	q1 Bf	q1 Br	v1	v2	v3	lambda
100	5000	2.00E-03	1.00E-14	250	3	200	3	0.854	0.84	0.82	0.001156
100	5000	2.00E-03	1.00E-14	250	3	200	3	0.854	0.84	0.0082	0.08984
100	5000	2.00E-03	1.00E-14	250	3	200	3	0.854	0.24	0.0082	0.9987
100	50000	2.00E-03	1.00E-14	250	3	200	3	0.854	0.24	0.0082	0.9981
100	5000	2.00E-04	1.00E-14	250	3	200	3	1.02	1	0.356	0.9987
100	500	2.00E-03	1.00E-14	250	3	200	3	1.02	0.1	0.439	0.9998
100	50	2.00E-08	1.00E-14	250	3	200	3	1.2	0.99	7.20E-05	0.045
100	5000	2.00E-08	1.00E-14	250	3	200	3	1.2	1	0.00001	0.9978

4.1 Results of homotopy after adding one resistor

The circuits having two resistors only could not give solutions even after so many attempts. So we decided to add one extra resistor between node 2 and node 3. And finally we got one solution i.e.

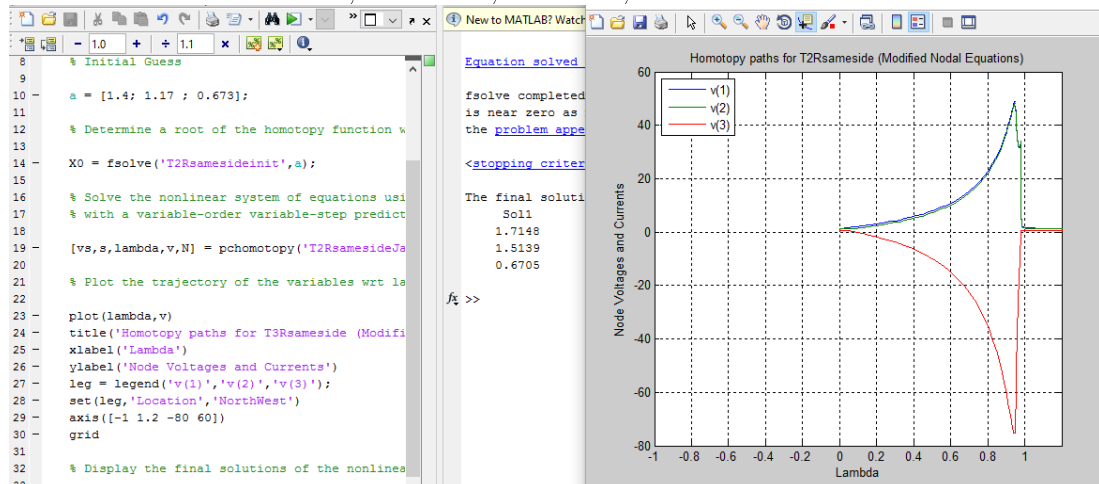
Sol1
 1.7148
 1.5139
 0.6705

Circuit diagram



Homotopy result:

Initial condition $v1=1.4$; $v2=1.17$; $v3=.693$; $i1=0.002$



5 Conclusion:

Based on the results that we got using homotopy applied over the above two circuits we conclude that homotopy is very sensitive to the initial condition. As described above in the section 1.2 convergent ball concept, to have a solution we must have initial condition in the ball. No solution in 2-transistor-2-resistor circuit after so many attempts and one solution in 2-transistor-3-resistor gives the insight that either the ball is too small so that we couldn't choose the initial condition inside that ball or the ball might not have existed for the 2-transistor-2-resistor circuit.

6 Modified Variable Stimulus Homotopy: Implementation.

We had already the fixed point homotopy code. We have a parser that can generate system nodal and modified nodal equations and the Jacobian of the equations. The equations and Jacobians are used in the implementation of homotopy.

6.1 Why do we need to implement this:

The types of homotopy functions described above are used to find the solutions of circuits having multiple dc operating points of the given circuits. And once we get the all dc operating points we can move on for the steady state analysis part. Since we may encounter various types of circuits and depending on the homotopy functions applied to find the solutions, we can get different convergence time. For example: variable gain homotopy is the fastest converging for bipolar circuits. So, we must have mathematical tool on which we can analyze the circuits. In this regard, I choose to implement modified variable stimulus homotopy.

6.2 Implementation:

$$\mathbf{H}(\mathbf{x}, \lambda) = (\mathbf{1} - \lambda)(\mathbf{x} - \mathbf{a})\mathbf{G}_{leak} + \mathbf{F}(\lambda\mathbf{x}) \quad (14)$$

In the Matlab code used in fixed point homotopy method we will use the predictor corrector method (pchomotopy) same as it is. But we need to change other Matlab functions appropriately. For the implementation part we will need these two equations also which are as follows:

$$\frac{dH}{d\lambda} = -G_{leak}(x - a) + Jacobian(\lambda x)x \quad (15)$$

And,

$$\frac{dH}{dx} = G_{leak}(1 - \lambda) + \lambda Jacobian(x) \quad (16)$$

NOTE: Order of multiplication of $Jacobian(\lambda x)$ and x in equation (15) must be maintained as both are matrix and non-commutative.

6.3 Finding constant a:

We have already discussed the importance of initial value in implementing the homotopy. Finding appropriate a is must for the convergence. Although the homotopy functions are globally convergent but when applying new homotopy initial value must be tweaked properly.

For our case we had initial value of fixed point homotopy and for this homotopy, our initial value will be calculated by following equation:

$$x_0 = a - \frac{F(0)}{G_{lk}} \quad (17)$$

This is because modified variable stimulus homotopy does not give initial value equal to 'a' unlike fixed point homotopy, as it does not reduce to $x=a$ as $\lambda=0$.

6.4 Points to remember while testing this homotopy on other circuits

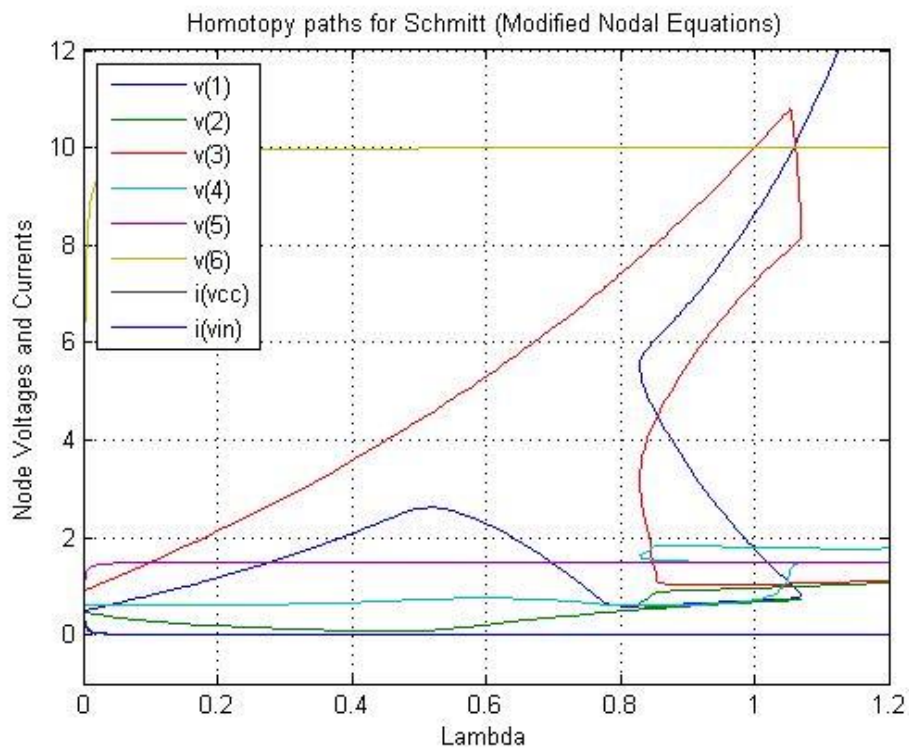
- Change a accordingly $x_0 + \frac{F(0)}{G_{lk}}$ by finding $F(0)$ ininit.m function by uncommenting "equation(lambda*X)"
- Paste equation and Jacobian at appropriate location in matlab functions
- In main.m function change the axis legend accordingly.
- Injac.m function change eye() as per the number of variables

7 Results using modified variable stimulus homotopy

The "modified variable stimulus homotopy" tested on the two circuits ie. Schmitt circuits and chua's circuit. And we got the following results which is shown below in the following table and graph:

7.1 Schmitt circuit (3 dc operating points)

Homotopy Path:



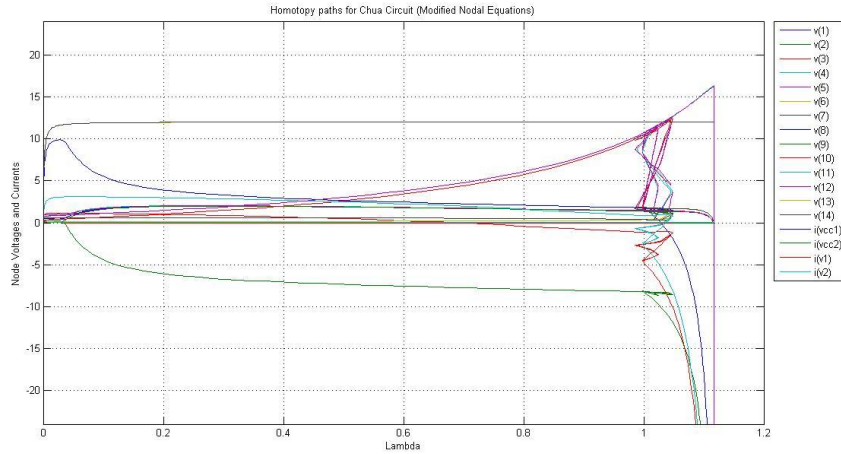
Homotopy Solution:

The final solutions for Schmitt (Modified Nodal Equations) are

Sol1	Sol2	Sol3
0.7103	1.7692	8.6257
0.6725	0.6918	0.9646
10.0000	7.2355	1.0383
0.7103	1.4900	1.7962
1.5000	1.5000	1.5000
10.0000	10.0000	10.0000
-0.0046	-0.0069	-0.0097
-0.0021	-0.0000	0.0000

7.2 Chua's circuit (9 dc operating points)

Homotopy Path:



Homotopy Solution:

The final solutions for Chua Circuit (Modified Nodal Equations) are

Sol1	Sol2	Sol3	Sol4	Sol5	Sol6	Sol7	Sol8	Sol9
1.7719	1.7835	1.7703	1.8344	1.8551	1.7834	1.7684	1.8268	1.7601
-8.2281	-8.2166	-8.2298	-8.1656	-8.1449	-8.2167	-8.2316	-8.1733	-8.2400
-1.1066	-2.5629	-2.9337	-4.4405	-4.3603	-2.5626	-2.9817	-4.4741	-4.7314
1.4640	7.7596	9.6007	8.8922	8.1419	1.4715	1.4615	7.9889	9.4817
1.4639	1.4717	1.4628	8.0358	8.1476	7.7596	9.8215	9.1853	9.6986
0.3687	1.8051	1.8230	1.8736	1.8833	0.3718	0.3693	1.8523	1.8097
0.3687	0.3707	0.3684	1.8608	1.8834	1.8051	1.8274	1.8713	1.8153
1.3882	1.4313	1.4423	1.4942	1.5063	1.3992	1.3848	1.4765	1.4297
1.3882	1.3994	1.3866	1.4847	1.5064	1.4312	1.4460	1.4910	1.4345
10.4580	3.3134	1.5424	1.6133	2.2040	10.2917	10.2424	2.5464	1.5299
0.8934	-0.5628	-0.9337	-2.4405	-2.3603	-0.5626	-0.9817	-2.4741	-2.7314
10.6933	10.5220	10.4784	2.6807	2.4294	3.5454	1.5440	1.6027	1.5328
12.0000	12.0000	12.0000	12.0000	12.0000	12.0000	12.0000	12.0000	12.0000
12.0000	12.0000	12.0000	12.0000	12.0000	12.0000	12.0000	12.0000	12.0000
-0.0030	-0.0032	-0.0032	-0.0034	-0.0034	-0.0031	-0.0031	-0.0034	-0.0032
-0.0030	-0.0030	-0.0030	-0.0033	-0.0034	-0.0032	-0.0032	-0.0033	-0.0032
-0.0007	-0.0006	-0.0005	-0.0004	-0.0004	-0.0006	-0.0005	-0.0004	-0.0004
0.0003	0.0004	0.0004	0.0002	0.0002	0.0001	0.0001	0.0001	0.0001

8 Implementation challenges in other homotopy

Having fixed point homotopy implemented already, I worked on implementing modified variable stimulus homotopy and got the same results. The other two homotopy “variable stimulus homotopy” and “variable gain homotopy” can also be implemented in the same way by finding $\frac{dH}{d\lambda}$, $\frac{dH}{dx}$ and x_0 appropriately.

The major problem in these homotopies will be the treatment of non-linear terms differently. Derivative of the homotopy function will require extra Matlab routine written according to the requirement which is not so difficult. But that will be very specific depending on circuits to circuits.

For now, we have parse which generate equations and Jacobian, only these two are used in the homotopy implemented. For other two homotopies Jacobian will require some terms in different format. And for those specific terms we will have to modify our parser in accordance to those equations as well.

9 References

- [1] L. Trajkovic, "DC Operating points of transistor circuits," *Nonlinear Theory and Its Applications, IEICE*, vol. 3, no. 3, pp. 287-300, July 2012.

- [2] Layne T. Watson, Stephen C. Billups and Alexander P. Morgan, *Algorithm 652, Hompack: A suite of Codes for Globally Convergent Homotopy Algorithms*.

- [3] Eula Fung, *Homotopy Methods for DC operating point and Periodic Steady- State Analysis: Spice3 Implementation*.

- [4] A. Dyess, *Finding dc operating points of chua's circuit using homotopy methods*, Technical report, University of Alabama, 1997.

- [5] A. Dyess, E. Chan, H. Hofmann, W. Horia, and L. Trajkovic, "Simple implementations of homotopy algorithms for finding dc solutions of nonlinear circuits," *Proceedings of the 1999 IEEE International Symposium on Circuits and Systems*, volume 6, pages 290–293 vol.6, Jul 1999.

- [6] L. Trajkovic and Alan N. Willson, Jr. *A Two-Transistor circuit having no resistors cannot model an SCR*. Research supported by NSF grants ECS 82-06207 and DCI 86-03639

- [7] L. Trajkovic and Alan N. Willson, Jr. *Behaviour of Non-linear Transistors One- Ports: Things are not simple as simple as might be expected*. Research supported by NSF grants ECS 82-06207 and DCI 86-03639