HOMSPICE: SIMULATOR WITH HOMOTOPY ALGORITHMS FOR FINDING DC AND STEADY-STATE SOLUTIONS OF NONLINEAR CIRCUITS

Ljiljana Trajković *

School of Engineering Science
Simon Fraser University
Burnaby, British Columbia
Canada V5A 1S6
ljilja@cs.sfu.ca

Eula Fung † *and Seth Sanders*

EECS Department
University of California
Berkeley, CA 94720-1770
eula@sequence.stanford.edu
sanders@eecs.berkeley.edu

ABSTRACT

We describe the use of homotopy (also called parameter embedding and continuation) methods for finding dc operating points and steady-state solutions of BJT and MOS transistor circuits. Past implementation of homotopy algorithms in proprietary industrial circuit simulators proved that they were viable options to resolving convergence difficulties for finding circuits' dc operating points. In this paper we describe a software implementation of publicly available homotopy algorithms (from the software package called HOMPACK) in the UCB SPICE circuit simulator. The new simulator, called **HomSPICE**, provides options for finding a circuit's dc operating points and steady-state solutions via three homotopy algorithms. We illustrate the performance of **HomSPICE** on several simulation examples.

1. INTRODUCTION

Parameter embedding methods are robust and accurate numerical techniques for solving nonlinear algebraic equations [1], [2]. They can be used to find multiple solutions of equations that possess more than one solution. A class of embedding algorithms called probability-one homotopy algorithms that promise global convergence [3] have been implemented in a publicly available software package HOMPACK [4]. Past research and implementations of homotopy algorithms for finding circuit dc operating points indicated promising results [5], [6], [7]. These algorithms have been used to find solutions to highly nonlinear circuits that could not be simulated using conventional numerical methods. They are also useful in finding dc operating points of multistable circuits. The main drawback of homotopy methods is their computational intensity. Therefore, they are most suitable for solving difficult nonlinear problems where initial solutions are hard to estimate, or when multiple solutions are desired. For circuits that fall in this category, homotopy algorithms offer a very attractive alternative.

We have implemented several homotopy algorithms in SPICE 3F5 [8], the latest version of "Berkeley SPICE" [9], [10]. The new simulator called **HomSPICE**, uses homotopy algorithms implemented in software package HOMPACK [4]. The three algorithms, FIXPDF, FIXPNF, and FIXPQF, are based on the ordinary differential equation, the normal flow, and the augmented Jacobian matrix methods, respectively. The advantage of using SPICE and HOMPACK is their publicly available source code. Further benefit of using SPICE [8] is that most commercial circuit simulators, being derived from SPICE, employ similar numerical algorithms for solving circuit equations.

Homotopy algorithms were added as options for two types of SPICE analyses: calculating a circuit's dc operating points and calculating a circuit's periodic steady-state response. Finding the dc operating point is essential for circuit simulation: both steady-state and transient analyses require a priori knowledge of a circuit's dc operating point. Finding a circuit's steady-state response is essential when designing power-conversion circuits, oscillators, and RF modulators. Many circuit performance measures, such as distortion and power dissipation, can only be measured accurately when the circuit has reached steady-state. Nevertheless, steady-state calculation is often not directly addressed by circuit simulators. In SPICE and most SPICE-like simulators, the steady-state response can only be found via a transient analysis. The steady-state is reached when transient effects become negligible. Unfortunately, this process may take an intolerably long time for lightly damped circuits, circuits with time constants much larger than their switching periods, and for high Q circuits. For such circuits, a direct steady-state solver is a better option. Two steady-state algorithms were implemented in SPICE 3C1 [11] and these routines have been included in **HomSPICE**.

This work was supported by the *NSF VPW Grant GER-9550153 and the †NSF Graduate Fellowship.

2. HOMOTOPY METHODS: BACKGROUND

Homotopy methods are used to solve systems of nonlinear algebraic equations and can be applied to a large variety of problems. We are most interested in solving the zero finding problem

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

where $\mathbf{x} \in \mathcal{R}^n$, $\mathcal{F} : \mathcal{R}^n \to \mathcal{R}^n$. (Note that the fixed point problem can be easily reformulated as a zero finding problem.)

We create the homotopy function $\mathcal{H}(\mathbf{x}, \lambda)$ by embedding $\mathcal{F}(\mathbf{x})$ into an equation of higher dimension

$$\mathcal{H}(\mathbf{x}, \lambda) = \mathbf{0}, \tag{2}$$

where $\lambda \in \mathcal{R}$, $\mathcal{H} : \mathcal{R}^n \times \mathcal{R} \to \mathcal{R}^n$. At $\lambda = 0$,

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

is an easy equation to solve, and at $\lambda = 1$,

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

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

A simple example of a homotopy is

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

where $\mathbf{a} \in \mathcal{R}^n$ is a known constant vector. Hence, $\mathcal{H}(\mathbf{x}, 0) := \mathbf{x} - \mathbf{a} = \mathbf{0}$ has an easy solution $\mathbf{x} = \mathbf{a}$, while $\mathcal{H}(\mathbf{x}, 1) := \mathcal{F}(\mathbf{x}) = \mathbf{0}$ is our original problem. By following solutions of $\mathcal{H}(\mathbf{x}, \lambda) = \mathbf{0}$ as λ varies from 0 to 1, we reach the solution to $\mathcal{F}(\mathbf{x}) = \mathbf{0}$.

The solutions trace a path known as the zero curve. Various numerical problems may occur depending on the behavior 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 [4], [2].

A variation [5] of the standard homotopy (5)

$$\mathcal{H}(\mathbf{x}, \lambda) = (1 - \lambda)\mathbf{G}(\mathbf{x} - \mathbf{a}) + \mathcal{F}(\lambda\mathbf{x}) \tag{6}$$

was implemented in the **HomSPICE** dc operating point analysis. Diagonal matrix $\mathbf{G}$ allows scaling of the linear term $(1 - \lambda)(\mathbf{x} - \mathbf{a})$ in (6). In **HomSPICE**, the elements of $\mathbf{G}$ were chosen as the coefficients of the corresponding x_i terms in $\mathcal{F}(\mathbf{x})$ with the idea that scaling the linear term might help convergence. For example, for the equation

$$h_i(\mathbf{x}, \lambda) = (1 - \lambda)g_{ii}(x_i - a_i) + f_i(\mathbf{x}), \tag{7}$$

g_{ii} is chosen to be equivalent to the coefficient of x_i in $f_i(\mathbf{x})$. In the case when the coefficient is too small ($< 10^{-8}$), causing the Jacobian matrix to lose rank, the g_{ii} value is set to 1.

3. DC OPERATING POINT ANALYSIS

Three steps were essential for implementing homotopy algorithms for finding a circuit's dc operating points:

- extract the nonlinear function $F(x)$ and its Jacobian from the data structure of the circuit simulator
- pass the information to the curve tracking algorithms, and
- create well behaved homotopies.

The first step was easily solved because SPICE [8] employs the Newton-Raphson method that requires knowing $\mathcal{F}(\mathbf{x})$ and its Jacobian. In the SPICE data structure, at every iteration (i), the Jacobian is explicitly saved in a matrix form, called here $\mathbf{A}$, along with a right hand side vector $\mathbf{b}$ that satisfies

$$\mathbf{A}|_{\mathbf{x}^{(i)}}\mathbf{x}^{(i)} - \mathcal{F}(\mathbf{x}^{(i)}) = \mathbf{b}|_{\mathbf{x}^{(i)}}. \tag{8}$$

Hence, at every iteration,

$$\mathcal{F}(\mathbf{x}) = \mathbf{A}\mathbf{x} - \mathbf{b}, \tag{9}$$

where $\mathcal{F}(\mathbf{x})$ are the Modified Nodal Analysis circuit equations [12].

Creating homotopies is rather straightforward, except for choosing which x_j in the linear term $(1 - \lambda)g_{ii}(x_j - a_i)$ needs to be added to each equation $f_i(\mathbf{x})$ when creating the homotopy (6). The proper choice is crucial or the Jacobian matrix of the homotopy may lose rank during curve tracking. While we theoretically know to which equation each variable should be added, we do not explicitly know the order of equations and variables in the simulator's data structure. In SPICE, the nodal current equations are written first, followed by the voltage equations. The variables are ordered with the voltage variables first followed by the current variables. However, the ordering of the current variables relative to the voltage variables is not transparent.

A simple, although not necessarily rigorous, algorithm for choosing the correct variable to add to each equation is to add x_j to the equation f_i where x_j has the largest coefficient among all the variables in $f_i(\mathbf{x})$. In case of Nodal Analysis equations written for node i, the variable x_i will have the only positive coefficient. In contrast, the Modified Nodal Analysis voltage equations contain only one nonzero coefficient, which corresponds to the variable we want to add when creating the homotopy function. In the Modified Nodal Analysis current equations, the largest coefficient usually corresponds to the current variable that we want to add. The algorithm implemented in **HomSPICE** uses a swapping technique to ensure that the same variable is never added twice.

The performance of **HomSPICE** is illustrated by simulating the triple Schmitt trigger cascade circuit [13]. The results are shown in Table 1.

4. STEADY-STATE ANALYSIS

Three standard techniques for computing the steady-state response directly are: finite difference, shooting, and harmonic balance [14]. The shooting method is the most attractive for implementation in SPICE. The finite difference method creates large systems of equations with many unknowns that can easily consume too much memory, and harmonic balance is a frequency domain technique, and, thus, does not take advantage of many of the features available in SPICE.

For a non-autonomous T-periodic system of the form

$$\dot{\mathbf{x}} = f(\mathbf{x}, t), \tag{10}$$

where $f(\mathbf{x}, t) = f(\mathbf{x}, t + T)$ for all t, a T-periodic steady-state solution $\tilde{\mathbf{x}}$ satisfies the two-point boundary condition

$$\tilde{\mathbf{x}}(0) = \tilde{\mathbf{x}}(T). \tag{11}$$

The solution to equation (10), subject to an initial condition constraint $\mathbf{x}(0) = \mathbf{x}_0$, can be expressed with the state transition function as $\mathbf{x}(t) = \phi(\mathbf{x}_0, 0, t)$, which is a function of time, initial state, and initial time. Any solution to the T-periodic steady-state problem will satisfy

$$\phi(\tilde{\mathbf{x}}(0), 0, T) = \tilde{\mathbf{x}}(0). \tag{12}$$

The shooting method focuses on solving equation (12), which can be viewed as searching for a fixed point of the map $\phi(\cdot, 0, T)$. The function $\phi(\cdot, 0, T)$ maps the state of the circuit starting from some initial state $\mathbf{x}_0$ at time 0 to its final state after the circuit variables have evolved for T seconds. The state variables are the currents through the inductors and voltages across the capacitors. The map $\phi(\cdot, 0, T)$ is called the *first return map* or *Poincaré map*.

The shooting problem can be solved by various methods, including Newton-Raphson [15] and extrapolation [16]. Both methods are locally convergent. Using homotopy methods promises global convergence, though the computation time will take longer. In both Newton-Raphson and globally convergent homotopies, the state transition function and its Jacobian matrix need to be computed for various values of the state variables. The state transition function at a given state x_0 is easy to compute using SPICE because it is equivalent to running a transient analysis on the circuit over one period using x_0 as the initial condition. The Jacobian matrix of $\phi(\cdot, 0, T)$ can be computed using sensitivity circuits. It is most efficiently computed during the transient analysis that already generates many required values. The details of calculating the Jacobian, otherwise known as the sensitivity matrix, can be found in [11] and [14].

To apply globally convergent homotopies to the shooting problem requires that the Poincaré map be:

- $\mathcal{C}^2$ and
- $\phi(\cdot, 0, T)$ maps a convex, compact set into itself.

However, we cannot guarantee that any of these conditions are met. For power electronic circuits which contain state-controlled switches, the Poincaré map may be everywhere differentiable, continuous and piecewise differentiable, or even discontinuous. In general, qualitative knowledge of the circuit's behavior should be sufficient to determine if the circuit will be an appropriate candidate for homotopy steady-state analysis.

The steady-state **HomSPICE** solutions for a dc power supply circuit [16] are shown in Table 2.

The two cycles in stead-state regime of each waveform are shown in Fig. 1 along with the solutions found by the homotopy analyses

The solution traces of the FIXPNF algorithm are shown in Fig. 2. The traces generated by the other algorithms are similar. The traces indicate the efficiently of the implemented homotopy for the given problem. Indeed, most of the curves are almost linearly approaching the solution, implying a good choice of homotopy.

5. CONCLUDING REMARKS

We have combined homotopy algorithms from the HOMPACK suite of globally convergent homotopy algorithms with the SPICE 3F5 circuit simulator, and have applied them successfully to finding dc operating points and steady-state solutions of transistor circuits. We used simple standard homotopy function that exploited the information readily available from the data structure of a circuit simulator. Our implementation, which employs public domain packages, illustrates that simple homotopies prove adequate for solving some difficult benchmark circuits. More sophisticated homotopies, such as "variable gain" (for BJT circuits) and "variable threshold" (for MOS circuits) promise even better performance.

6. REFERENCES

[1] E. L. Allgower and K. Georg, *Numerical Continuation Methods: An Introduction.* New York: Springer-Verlag Series in Computational Mathematics, 1990, pp. 1–15.

[2] L. T. Watson, "Globally convergent homotopy algorithms for nonlinear systems of equations," *Nonlinear Dynamics,* vol. 1, pp. 143–191, Feb. 1990.

[3] S. Chow, J. Mallet-Paret, and J. A. Yorke, "Finding zeroes of maps: homotopy methods that are constructive with probability one," *Mathematics of Computation,* vol. 32, no. 143, pp. 887–899, July 1978.

[4] L. T. Watson, S. C. Billups and A. P. Morgan, "Algorithm 652: HOMPACK: A suite of codes for globally

Solution	FIXPDF		FIXPNF	
	No. of Jacobian eval.	λ	No. of Jacobian eval.	λ
1.	918	1.000004	250	1.000000
2.	1867	1.000000	450	0.999983
3.	2121	1.000001	508	1.000000

Table 1: The number of Jacobian evaluations required for finding each operating point by using FIXPDF and FIXPNF for the triple Schmitt trigger cascade circuit.

State variable	Transient	FIXPDF	FIXPNF	FIXPQF
c1	-1.759777e+01	-1.779588e+01	-1.778526e+01	-1.778065e+01
c2	1.775704e+01	1.776682e+01	1.775577e+01	1.775117e+01
c3	1.787620e+01	1.788487e+01	1.787421e+01	1.787068e+01
l1	1.767396e-02	1.773145e-02	1.769987e-02	1.772588e-02
No. of Jacobian eval.		32	13	10
λ		1.000124	1.000094	0.999803

Table 2: The result of the transient analysis at time 1.98 seconds, and the steady-state initial conditions found by **HomSPICE** for the dc power supply circuit [16].

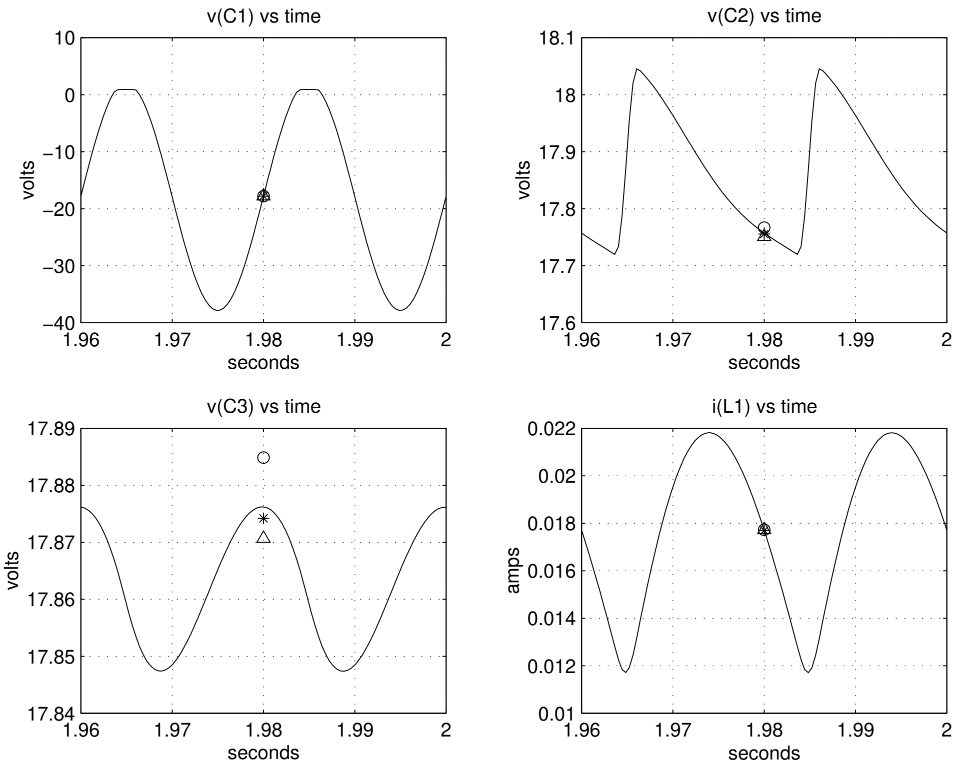

Figure 1: The steady-state initial conditions for the dc power supply [16] found by the homotopy methods plotted against the circuit's steady-state waveforms: $\circ$ are the FIXPDF solutions, $*$ are the FIXPNF solutions, and $\triangle$ are the FIXPQF solutions.

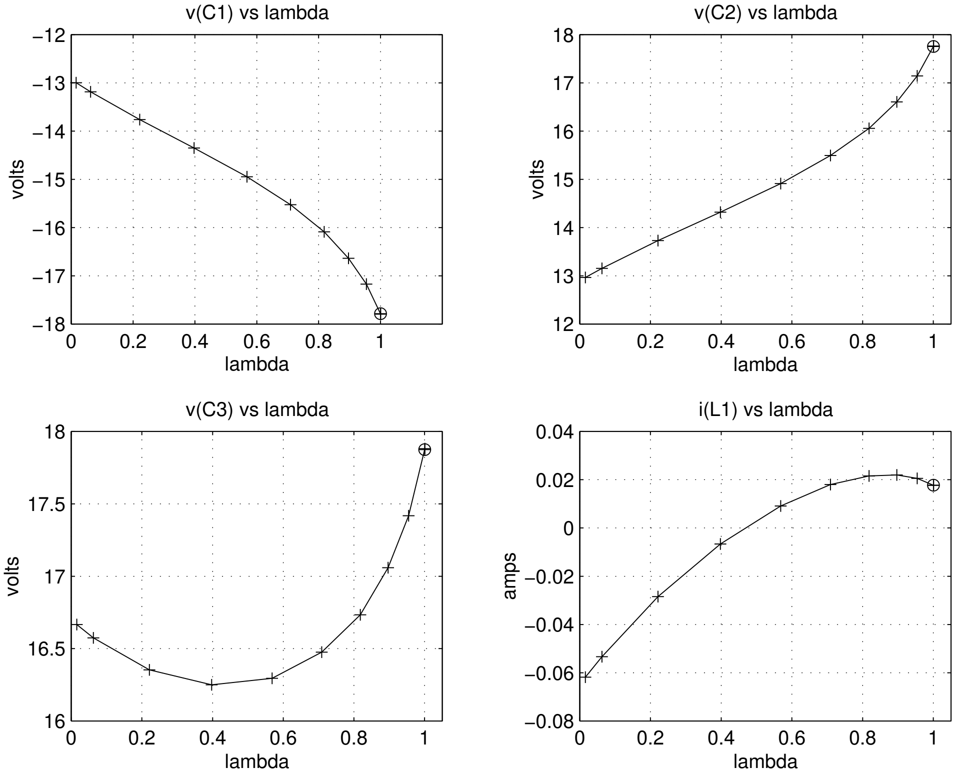

Figure 2: The solution curves for the dc power supply [16] generated by the FIXPNF algorithm.

convergent homotopy algorithms," *ACM Trans. Mathematical Software*, vol. 13, no. 3, pp. 281–310, Sept. 1987.

[5] R. C. Melville, Lj. Trajković, S. C. Fang and L. T. Watson, "Artificial parameter homotopy methods for the dc operating point problem," *IEEE Trans. on CAD,* vol. 12, no. 6, pp. 861–877, June 1993.

[6] Lj. Trajković and W. Mathis, "Parameter embedding methods for finding dc operating points: Formulation and implementation," *Proc. NOLTA '95*, Las Vegas, NV, Dec. 1995, pp. 1159–1164.

[7] D. Wolf and S. Sanders, "Multiparameter homotopy methods for finding dc operating points of nonlinear circuits," *IEEE Trans. Circuits Syst.,* vol. 43, pp. 824–838, Oct. 1996.

[8] T. L. Quarles, A. R. Newton, D. O. Pederson, and A. Sangiovanni-Vincentelli, "SPICE 3 Version 3F5 User's Manual," Department of EECS, University of California, Berkeley, March 1994.

[9] T. L. Quarles, "The SPICE3 Implementation Guide," *Memorandum No. UCB/ERL M89/44*, Department of EECS, University of California, Berkeley, April 24, 1989.

[10] A. Vladimirescu, *The SPICE Book*. New York: John Wiley & Sons, Inc., 1994.

[11] P. N. Ashar, "Implementation of algorithms for the periodic steady-state analysis of nonlinear circuits," *M. S. Thesis*, UC Berkeley, March 1989.

[12] C. W. Ho, A. E. Ruehli, and P. A. Brennan, "The modified nodal approach to network analysis," *IEEE Trans. Circuits Syst.*, vol. CAS-22, pp. 504–509, Jan. 1975.

[13] P. Horowitz and W. Hill, *The Art of Electronics.* Cambridge, MA: Cambridge University Press, 1983.

[14] K. S. Kundert, J. K. White, and A. Sangiovanni-Vincentelli, *Steady-State Methods for Simulating Analog and Microwave Circuits.* Boston, MA: Kluwer Academic Publishers, 1990.

[15] T. J. Aprille and T. N. Trick, "Steady-state analysis of nonlinear circuits with periodic inputs," *Proc. IEEE*, vol. 60, pp. 108–14, January 1972.

[16] S. Skelboe, "Computation of the periodic steady-state response of nonlinear networks by extrapolation methods," *IEEE Trans. Circuits Syst.*, vol. CAS-27, pp. 161–175, March 1980.