Numerical Methods in DC Circuit Analysis

Edward Hayes

April 17, 2022

Abstract

Many electronic devices exhibit a nonlinear current-voltage relationship. During circuit design and analysis, this nonlinearity makes traditional circuit analysis techniques insufficient for solving for node voltages and currents. An iterative approach is used. The Newton-Raphson method with numerical differentiation was used to solve a simple nonlinear circuit. The solution to the nonlinear circuit was consistent with the result of professional circuit software.

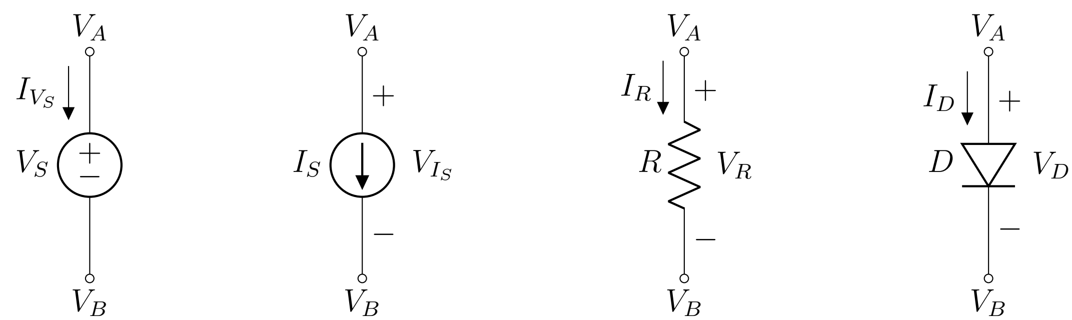
1 Introduction

Traditional circuit analysis techniques are inadequate for the analysis of nonlinear circuits. In order to analyze a nonlinear circuit, numerical methods are used to iteratively solve for circuit variables such as the voltages and currents within the circuit. SPICE, or "Simulation Program with Integrated Circuit Emphasis", is often used to solve linear and nonlinear circuits. This paper will outline the mathematics used to solve for voltages of nonlinear circuits under direct current, or DC, conditions. Note that this is only a subset of the capabilities that SPICE software offers.

The focus of this paper is on the numerical methods used in circuit analysis[4]. As a result, the principles of electricity and circuit analysis will not be expressed in its entirety. The following examples will illustrate the concepts needed for this paper. The basics of circuit analysis, including the concepts in this paper, can be found in [1].

Let us start by examining the relationship between voltage and current of circuit components with an example.

Example 1. Plot the current-voltage (IV) characteristic curves for the components shown in Figure 1.



(a) Voltage source. (b) Current source. (c) Resistor. (d) Diode.

Figure 1: Ideal circuit components.

The relationship between current and voltage can best be summarized using Ohm's law. Ohm's law states that the current flowing through a conductor is proportional to the voltage across is. The equation for Ohm's law is,

$$I = \frac{V}{R} \tag{1}$$

where I is the current measured in amps (A), V is the voltage measured in volts (V), and R is the resistance, or the constant of proportionality, measured in ohms (Ω). It is important to note that Ohm's law is not always obeyed. For the components in Figure 1, the relationship between current and voltage are as follows:

1. A voltage source produces a constant voltage, V_S, no matter the current flowing through it. Thus, the current through a voltage source is unknown.

2. A current source produces a constant current, I_S, no matter the voltage across it. Thus, the voltage across a voltage source is unknown.

3. Resistors obey Ohm's law. Using Equation (1), the current through a resistor is,

$$I_R = \frac{V_R}{R} = \frac{V_A - V_B}{R} \tag{2}$$

   where V_A and V_B are the positive and negative terminal voltages, respectively.

4. Diodes do not obey Ohm's law. The current through a diode is,

$$I_D = I_{SAT}\left[\exp\left(\frac{qV_D}{\eta kT}\right) - 1\right] \tag{3}$$

   where I_{SAT} is the reverse saturation current, q is the charge of an electron, η is the non-ideality factor, k is Boltzmann's constant and T is the absolute temperature in degrees Kelvin. The parameters I_{SAT} and η are typically given in the datasheet for a specific diode. Refer to [1] and [4] for diode modeling.

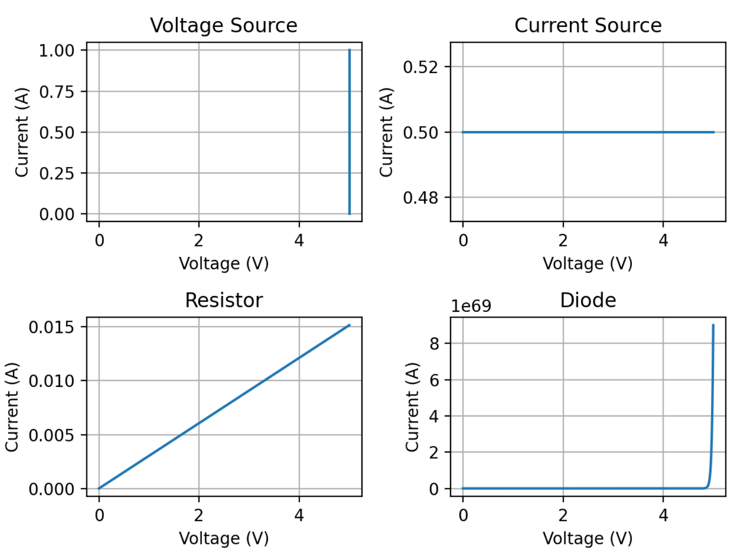


Figure 2: IV characteristic curves for components in Figure 1.

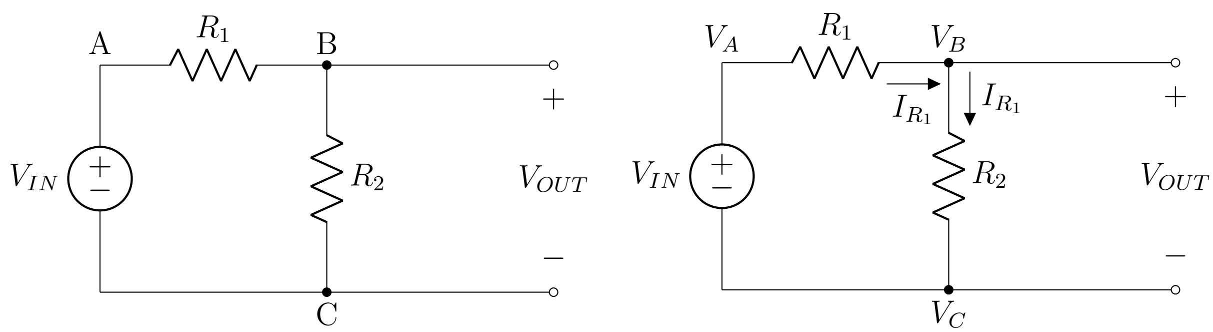
The component IV curves are shown in Figure 2. The parameters used for producing the plots are given in Table 1.

Table 1: Plot Parameters

Parameter	Value
Voltage Source, V_S	5V
Current Source, I_S	0.5A
Resistance, R	330Ω
Resistor Voltage, V_R	$0 - 5$V
Diode Voltage, V_D	$0 - 5$V
Reverse Saturation Current, I_{SAT}	1×10^{-14}A
Electron Charge, q	$1.60217663 \times 10^{-19}$C
Non-Ideality Factor, η	1
Boltzmann's Constant, k	1.380649×10^{-23}J/K
Temperature, T	300.15K

It can be seen in Figure 2 that ideal sources and resistors are linear components. Conversely, diodes are nonlinear components. Therefore, a circuit can be defined as nonlinear if it contains at least one nonlinear component. Traditional circuit analysis techniques depend on linearity between voltage and current. A demonstration of circuit analysis on a simple linear circuit is given in the example below.

Example 2. Determine the node voltages for the given circuit when $V_{IN} = 5$V and $R_1 = R_2 = 330\Omega$.



(a) Circuit with node labels. (b) Circuit with node voltages and currents.

Figure 3: Linear circuit.

Start by labeling the node voltages and currents of the circuit shown in 3a. The result is shown in figure 3b. To determine node voltages, Kirchhoff's current law is used to relate node currents to node voltages. Kirchhoff's current law (KCL) states that the sum of currents entering and exiting a node is zero. By convention, currents entering a node are negative and currents exiting a node are positive. The equation for KCL is,

$$\sum_{k=1}^{n} i_k = 0. \tag{4}$$

Now to determine nodal equations, start by designating a reference, or ground node. Let Node C be the ground node.

Thus,

$$V_C = 0\text{V} \tag{5}$$

As a result,

$$V_A = V_{IN} \tag{6}$$
$$V_B = V_{OUT} \tag{7}$$

Applying KCL at Node B results in,

$$-I_{R_1} + I_{R_2} = 0 \tag{8}$$

Using Equation (2), Equation (8) becomes,

$$\frac{V_B - V_A}{R_1} + \frac{V_B}{R2} = 0 \tag{9}$$

Combining Equations (6), (7) and (9),

$$\frac{V_{OUT} - V_{IN}}{R_1} + \frac{V_{OUT}}{R2} = 0 \tag{10}$$

Then solving for V_{OUT} gives,

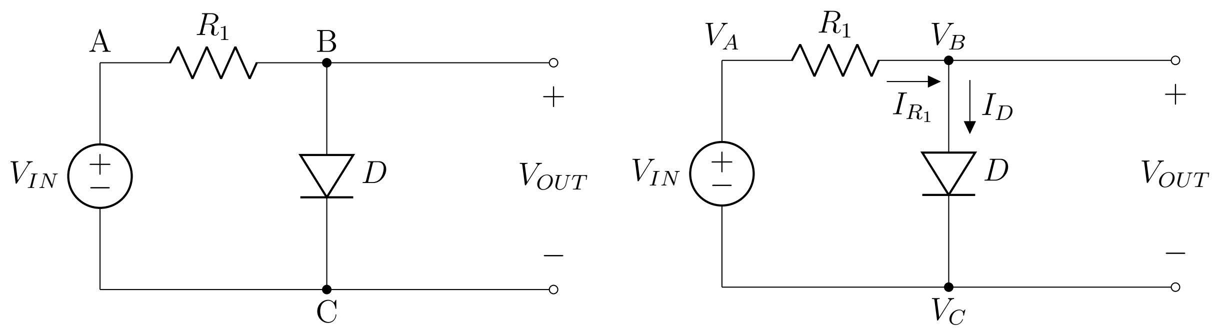
$$V_{OUT} = V_{IN}\left(\frac{R_2}{R_1 + R_2}\right) \tag{11}$$

The solved node voltages are:

$$V_A = 5\text{V}$$
$$V_B = 2.5\text{V}$$
$$V_C = 0\text{V}$$

Now replace the resistor R_2 with a diode and perform circuit analysis on this nonlinear circuit.

Example 3. Determine the node voltages for the given circuit when $V_{IN} = 5\text{V}$ and $R_1 = 330\Omega$.



(a) Circuit with node labels. (b) Circuit with node voltages and currents.

Figure 4: Nonlinear circuit.

Start by labeling the node voltages and currents of the circuit shown in 4a. The result is shown in figure 4b. Now to determine nodal equations, start by designating a reference, or ground node. Let Node C be the ground node.

Thus,

$$V_C = 0\text{V} \tag{12}$$

As a result,

$$V_A = V_{IN} \tag{13}$$
$$V_B = V_D = V_{OUT} \tag{14}$$

Applying KCL at Node B results in,

$$-I_{R_1} + I_D = 0 \tag{15}$$

Using Equation (2) and Equation (3), Equation (15) becomes,

$$\frac{V_B - V_A}{R_1} + I_{SAT}\left[\exp\left(\frac{qV_D}{\eta kT}\right) - 1\right] = 0 \tag{16}$$

Combining Equations (13), (14) and (16),

$$\frac{V_{OUT} - V_{IN}}{R_1} + I_{SAT}\left[\exp\left(\frac{qV_{OUT}}{\eta kT}\right) - 1\right] = 0 \tag{17}$$

Note that Equation (17) is a nonlinear equation in the form,

$$f(V_{OUT}) = \frac{V_{OUT} - V_{IN}}{R_1} + I_{SAT}\left[\exp\left(\frac{qV_{OUT}}{\eta kT}\right) - 1\right] = 0 \tag{18}$$

Thus, V_{OUT} must be solved for iteratively.

Example 3 shows that nonlinear circuits require numerical methods. In the next section the numerical methods required to solve this circuit will be discussed.

2 Theory

In order to solve the nonlinear equation, Equation (18), in Example 3, the Newton-Raphson method will be used in conjunction with numerical differentiation.

2.1 Newton-Raphson Method

The Newton-Raphson method[2] is an algorithm for finding the root, x_r, of a smooth and continuous function, $f(x)$. The root of a function is defined as $f(x_r) = 0$. The method starts with a guess, x_n. Assuming that $x_n \neq x_r$ but x_n is "close" to x_r, an improved guess, x_{n+1}, can be made. In order to make this improved guess, a linear approximation of $f(x)$ at x_n is taken using a first-order Taylor polynomial,

$$f_{approx}(x) = f(x_n) + f'(x_n)(x - x_n). \tag{19}$$

Using Equation (19), the next guess is x_{n+1} such that $f_{approx}(x_{n+1}) = 0$,

$$0 = f_{approx}(x_{n+1}) \tag{20}$$
$$0 = f(x_n) + f'(x_n)(x_{n+1} - x_n) \tag{21}$$
$$x_{n+1} = x_n - \frac{f(x_n)}{f'(x_n)} \tag{22}$$

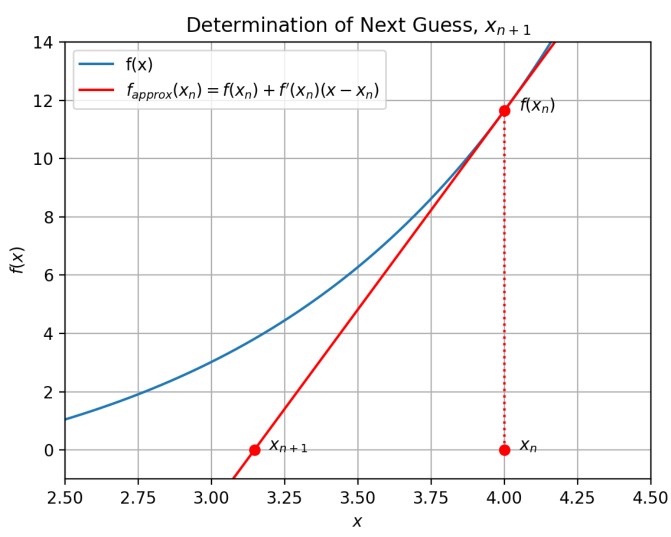


Figure 5: Linear approximation at x_n.

The process of linearizing $f(x)$ about x_n is shown in Figure 5. This is also referred to as a Newton step. Additional guesses, or steps, will be made until the k^{th} guess is sufficiently close enough to x_r such that,

$$f(x_k) < \varepsilon \tag{23}$$

where ε is the allowable error, or tolerance. At which time, it is said that the method converges to x_r. Note that depending on the initial guess, x_n, the method may converge to a different root than, x_r. Also,the Newton steps may vary depending on the derivative of $f(x)$. They may be very large and leading away from the root, x_r. Below is an example of the Newton-Raphson method.

Example 4. Find the root of the following function.

$$f(x) = \frac{1}{4}e^x - 2 \tag{24}$$

Start by taking the derivative of $f(x)$,

$$f'(x) = \frac{1}{4}e^x \tag{25}$$

Let the initial guess be $x_0 = 4$. First check if x_0 is a root.

$$f(x_0) = f(4) = 11.650 \neq 0 \tag{26}$$

Since x_0 is not a root, find the next guess using the linearization of $f(x)$.

$$x_1 = x_0 - \frac{f(x0)}{f'(x_0)} = 4 - \frac{11.650}{13.650} = 3.147 \tag{27}$$

Check if x_1 is a root.

$$f(x_1) = f(3.147) = 3.814 \neq 0 \tag{28}$$

Since x_1 is not a root, this process will continue until x_k is sufficiently close to a root. This iterative process is illustrated in Figure 6.

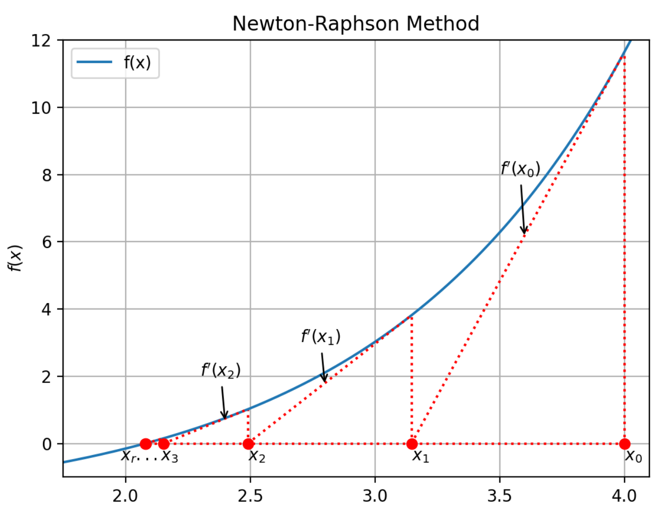


Figure 6: Iterations for $f(x)$.

Eventually the root is found to be $x_5 = 2.079445$ which is very close to the actual root $x_r = \ln(8)$.

2.2 Numerical Differentiation

A derivative is the slope of a tangent line at some point of a function, $f(x)$. For the point $x = a$, the derivative is defined as,

$$f'(a) = \lim_{x \to a} \frac{f(x) - f(a)}{x - a}. \tag{29}$$

In order to numerically approximate the slope of this tangent line, points about $x = a$ will be used. This is called a finite difference approximation[3]. Specifically, the Central Difference will be discussed. The Central Difference uses two adjacent points with equal distance, h, from $x = a$.

Let x_j be the point of interest. The two adjacent points are then, $(x_{j-1}, f(x_{j-1}))$ and $(x_{j+1}, f(x_{j+1}))$. The Central Difference is then given by,

$$f'(x_j) = \frac{f(x_{j+1}) - f(x_{j-1})}{x_{j+1} - x_{j-1}}. \tag{30}$$

Since the points are $\pm h$ apart from x_j, meaning a difference of $2h$, the Central Difference can be rewritten as,

$$f'(x_j) = \frac{f(x_j + h) - f(x_j - h)}{2h}. \tag{31}$$

Note that as h decreases, the approximation gets closer to the actual derivative at x_j. An example of the Central Difference is given below.

Example 5. Plot the Central Difference approximations at $x = \frac{1}{4}$ for step sizes $h = 0.01, 0.1, 0.2$ for the following function.

$$f(x) = \frac{1}{2}x^3 + 2x^2 - 1x + 1 \tag{32}$$

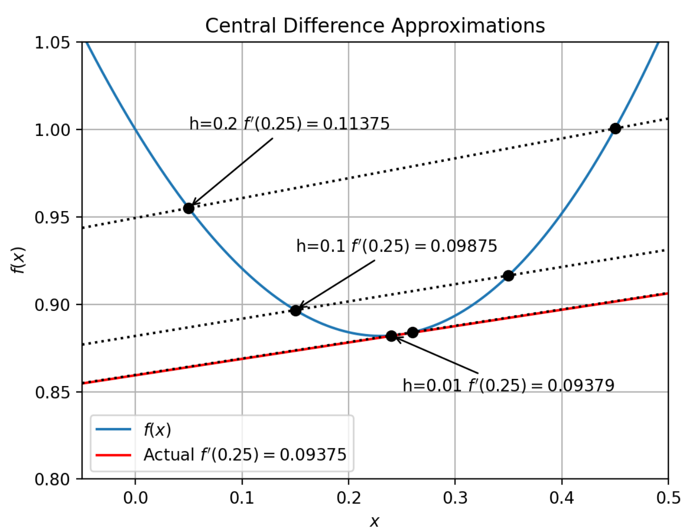


Figure 7: Central difference approximations for $f(x)$.

As seen in Figure 7, the approximation improves as the step size, h, decreases.

3 Implementation

The applications of the numerical methods in the previous section can easily be accomplished with Python. Below is the Python implementations of the Newton-Raphson method, Central Difference method and the solution to Equation (18). Note that the solution is the final node voltage for the nonlinear circuit in Example 3. The Python code utilizes the following:

1. The scientific computing package, NumPy, for exponential functions.
2. The Central Difference method with Equation (31). The default step size is $h = 0.001$.
3. The Newton-Raphson method with Equation (22). The tolerance is $\varepsilon = 1 \times 10^{-6}$.
4. Circuit parameters from Table 1 and Example 3.
5. Nonlinear equation, Equation (18), which represents the output voltage for the nonlinear circuit.

```
# import numpy library for exponential function
import numpy as np

# central difference approximation for numerical differentiation
def central_diff(f,x0,h=0.001):
    return (f(x0+1*h)-f(x0-1*h))/(2*h)

# newton-raphson method for root finding
def newton_raphson(f,x0,tol,count=0):

    #print current iteration count and x_n
    print('Iteration: ',count,' Value: ',x0)

    # check if x_n is a root
    if abs(f(x0))<tol:
        return x0

    # if x_n is not a root, determine x_n+1
    # and repeat another iteration
    else:
        x1 = x0 - f(x0)/central_diff(f,x0)
        return newton_raphson(f,x1,tol,count+1)

# constants for nodal equation
VIN = 5               # input voltage (V)
R = 330               # resistor value (Ohms)
ISAT = 1e-14          # reverse saturation current (A)
Q = 1.60217663e-19    # charge of an electron (C)
N = 1                 # non-ideality factor
K = 1.380649e-23      # boltzmanns constant (J/K)
T = 300.15            # absolute temperature (K)

# nodal equation
```

```python
f = lambda Vout: (Vout-VIN)/R+ISAT*(np.exp((Q*Vout)/(N*K*T))-1)

# solve for root (Vout) of the nodal equation
root = newton_raphson(f,0,1e-6)

# print result
print('\nVout: ',root)
```

Listing 1: Python Solution for Nonlinear Circuit.

After running the above code, the algorithm converges to $V_{OUT} = 0.7213915764512933$ on the 170^{th} iteration. Thus for completeness, the node voltages in Example 3 are:

$$
\begin{aligned}
V_A &= 5\text{V} \\
V_B &= 0.7213915764512933\text{V} \\
V_C &= 0\text{V}
\end{aligned}
$$

Using the popular circuit software LTspice, V_{OUT} was given as 0.721392V. For the purpose of this analysis, these values are essentially identical.

4 Conclusion

In conclusion, the Newton-Raphson method for root finding and numerical differentiation are required when analyzing nonlinear circuits. Example 3 is an example of a circuit with only one unknown circuit variable. That circuit variable, voltage is this case, was found using the Newton-Raphson method with numerical differentiation. The solution in this paper was consistent with the result from the professional circuit software, LTspice.

In a nonlinear circuit with multiple unknown variables, a system of nonlinear equations will need to be solved for simultaneously. In order to solve a system of nonlinear equations, a multivariable Newton-Raphson method is used with a Jacobian matrix. Linear algebra techniques are used with each iteration of the multivariable Newton-Raphson method until the nonlinear equations are solved. The circuit can then be solved using normal circuit analysis techniques with the solutions from the system of nonlinear equations. Multiple node circuits are much more complicated but the numerical methods in this paper translate to these problems as well.

References

[1] J.D. Irwin and R.M. Nelms. *Basic Engineering Circuit Analysis, 11th Edition*. Wiley, 2015.

[2] Qingkai Kong, Timmy Siauw, and Alexandre M. Bayen. Chapter 19 - root finding. In Qingkai Kong, Timmy Siauw, and Alexandre M. Bayen, editors, *Python Programming and Numerical Methods*, pages 325–335. Academic Press, 2021.

[3] Qingkai Kong, Timmy Siauw, and Alexandre M. Bayen. Chapter 20 - numerical differentiation. In Qingkai Kong, Timmy Siauw, and Alexandre M. Bayen, editors, *Python Programming and Numerical Methods*, pages 337–351. Academic Press, 2021.

[4] Lawrence Pillage. *Electronic Circuit & System Simulation Methods (SRE)*. McGraw-Hill, Inc., USA, 1 edition, 1998.