Modified Nodal Analysis - printable

This page is a collection of all of the modified nodal analysis pages that are collected into one document for easy printing.

Analysis of Circuits - Intro

Introduction | MNA Basics | Algorithmic MNA | Examples | Reacitve Circuits and OpAmps | SCAM | Dependent Sources | Printable

The following text is broken into several sections. Most are simply explanatory. You may skip directly to SCAM, a MATLAB® tool for deriving and solving circuit equations symbolically if you are not interested in the theory.

Contents

Introduction

Solving a set of equations that represents a circuit is straightforward, if not always easy. However, developing that set of equations is not so easy. The two commonly taught methods for forming a set of equations are the node voltage (or nodal) method and the loop-current (or mesh) method. I will briefly describe each of these, and mention their benefits and disadvantages. If you are interested in a more thorough introduction to these methods, I suggest that you do a web search for an appropriate tutorial. I will end with a discussion of a third method, Modified Nodal Analysis, that has some unique benefits. Among its benefits is the fact that it lends itself to algorithmic solution -- the ultimate goal of these pages is to describe how to use a MATLAB program for generating a set of equations representing the circuit that can be solved symbolically. If you are only interested in using that program you may go directly to the page using the tabs above.

Circuits discussed herein are simple resistive circuits with independent voltage and current sources. Dependent sources can be added in a straightforward way, but are not considered here.

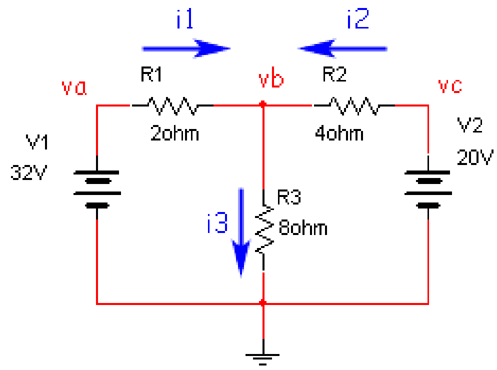
Node Voltage Method

To apply the node voltage method to a circuit with *n* nodes (with *m* voltage sources), perform the following steps (after Rizzoni).

1. Selective a reference node (usually ground).
2. Name the remaining n-1 nodes and label a current through each passive element and each current source.
3. Apply Kirchoff's current law to each node *not* connected to a voltage source.
4. Solve the system of *n-1-m* unknown voltages.

Example 1

Consider the circuit shown below



Steps 1 and 2 have already been applied. To apply step 3:

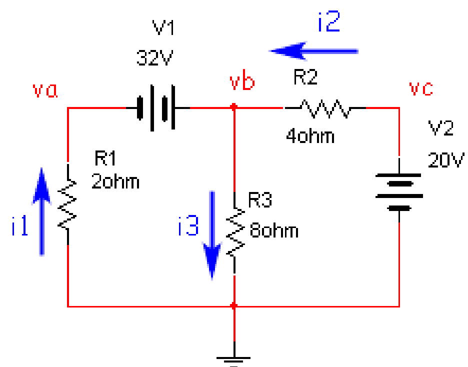
$$i_1 + i_2 = i_3$$
$$i_1 = \frac{v_a - v_b}{R_1}, \quad i_2 = \frac{v_c - v_b}{R_2}, \quad i_3 = \frac{v_b}{R_3}$$
$$v_a = V_1, \quad v_c = V_2$$
$$so$$
$$\frac{v_a - v_b}{R_1} + \frac{v_c - v_b}{R_2} = \frac{v_b}{R_3}$$
$$\frac{V_1 - v_b}{R_1} + \frac{V_2 - v_b}{R_2} = \frac{v_b}{R_3}$$

In this case there is only one unknown; plugging in numbers and solving the circuit we get $v_b = 24$.

The node-voltage method is generally straightforward to apply, but becomes a bit more difficult if one or more of the voltage sources is not grounded.

Example 2

Consider the circuit shown below.



Clearly this circuit is the same as the one shown above, with V1 and R_1 interchanged. Now we write the equations:

$$i_1 + i_2 = i_3$$
$$i_1 = \frac{-v_a}{R_1}$$
$$i_2 = \frac{v_c - v_b}{R_2}$$

$$i_3 = \frac{v_b}{R_3}$$

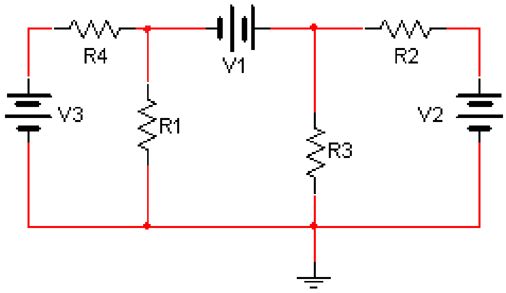
The difficulty arises because the voltage source V_1 is no longer identical to one of the node voltages. Instead we have

$$\begin{aligned} v_b - v_a &= V_1 \\ v_c &= V_2 \\ &or \\ v_a &= v_b - V_1 \\ &so \\ \frac{-v_a}{R_1} + \frac{v_c - v_b}{R_2} &= \frac{v_b}{R_3} \\ \frac{V_1 - v_b}{R_1} + \frac{V_2 - v_b}{R_2} &= \frac{v_b}{R_3} \end{aligned}$$

Note that the last line is the same as that from the previous circuit, but to solve the circuit we had to first solve for v_a. This procedure wasn't difficult, but required a little cleverness, and will be a bit different for each circuit layout. Another way to handle this problem is to use the concept of a supernode, which complicates the rules for setting up the equations (DeCarlo/Lin). However, the supernode concept handles the case of a non-grounded voltage source without any need for solving intermediate equations, as we did here.

The examples chosen here were simple but illustrated the basic techniques of nodal analysis. It also illustrated one of the difficulties with the technique, setting up equations with a floating voltage source. The technique of modified nodal analysis, introduced later, also has no difficulties when presented with floating voltage sources.

In the case shown above it would be possible to create a solution by simply reversing the order of R_1 and V_1. However we are trying to develop an algorithmic technique so we don't need to apply insights that might be hard to program. Additionally, a circuit like the one below (with V_1 ungrounded), has no such obvious simplifications.



Mesh Current Method

The mesh current method is, not surprisingly, similar to the node voltage method. We will define a "mesh" as an open "window" in the schematic drawing. *Note that this method requires some modification to work with "non-planar" circuits (non-planar circuits are those that cannot be drawn without wires that cross each other without touching).* The rules below follow those in Rizzoni.

To apply the loop current method to a circuit with *n* meshes (and with *m* current sources), perform the following steps.

1. Define each mesh and its corresponding current. This is easiest with a consistent method, e.g. all unknown currents are clockwise, all know currents follow direction on current source.
2. Apply Kirchoff's voltage law to each loop *not* containing a current source.
3. Solve the system of *n-m* unknown voltages.

Example 3

Consider the circuit from Example 1, with mesh currents defined.

Modified Nodal Analysis - Intro

Contents

Introduction

Though the node voltage method and loop current method are the most widely taught, another powerful method is modified nodal analysis (MNA). MNA often results in larger systems of equations than the other methods, but is easier to implement algorithmically on a computer which is a substantial advantage for automated solution. To use modified nodal analysis you write one equation for each node not attached to a voltage source (as in standard nodal analysis), and you augment these equations with an equation for each voltage source. To be more specific, the rules for standard nodal analysis are shown below:

Node Voltage Method

To apply the node voltage method to a circuit with *n* nodes (with *m* voltage sources), perform the following steps (after Rizzoni).

1. Select a reference node (usually ground).
2. Name the remaining n-1 nodes and label a current through each passive element and each current source.
3. Apply Kirchoff's current law to each node *not* connected to a voltage source.
4. Solve the system of *n-1-m* unknown voltages.

The difficulty with this method comes from having to consider the effect of voltage sources. Either a separate equation is written for each source, or the supernode method must be used.

The rules for modified nodal analysis are given by:

Modified Nodal Analysis

To apply the node voltage method to a circuit with *n* nodes (with *m* voltage sources), perform the following steps (after DeCarlo/Lin).

1. Selective a reference node (usually ground) and name the remaining n-1 nodes. Also label currents through each current source.

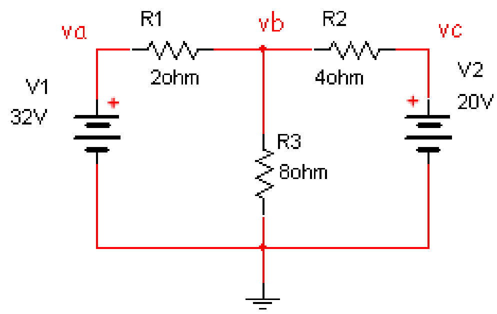
2. Assign a name to the current through each voltage source. We will use the convention that the current flows from the positive node to the negative node of the source.
3. Apply Kirchoff's current law to each node. We will take currents out of the node to be positive.
4. Write an equation for the voltage each voltage source.
5. Solve the system of *n-1* unknowns.

Note: I will only discuss independent current and voltage sources. Dependent sources are a simple extension. See Litovski or DeCarlo/Lin for reference.

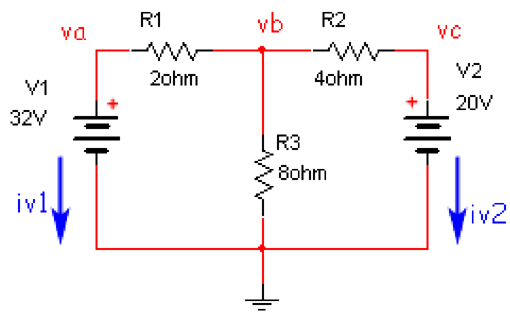
As an example consider the circuit below (from the previous document)

Example 1

Consider the circuit shown below (Step 1 has already been applied)



Apply step 2 (currents through the voltage sources with current from positive node to negative node):



Apply step 3 (with positive currents out of the node):

$$Node\ a: \quad i_{v1} + \frac{v_a - v_b}{R_1} = 0$$

$$Node\ b: \quad \frac{v_b - v_a}{R_1} + \frac{v_b}{R_3} + \frac{v_b - v_c}{R_2} = 0$$

$$Node\ c: \quad i_{v2} + \frac{v_c - v_b}{R_2} = 0$$

Apply step 4:

$$v_a = V_1$$
$$v_c = V_2$$

Apply step 5:

$$
\begin{aligned}
i_{v1} + \frac{v_a - v_b}{R_1} &= 0 \\
\frac{v_b - v_a}{R_1} + \frac{v_b}{R_3} + \frac{v_b - v_c}{R_2} &= 0 \\
i_{v2} + \frac{v_c - v_b}{R_2} &= 0 \\
v_a &= V_1 \\
v_c &= V_2
\end{aligned}
$$

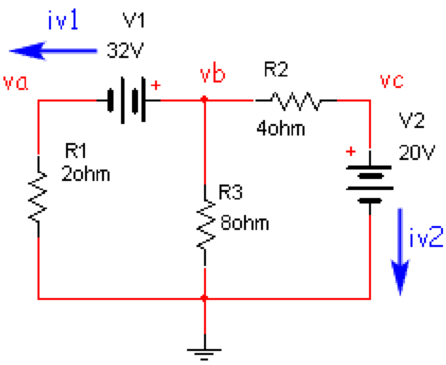
or

$$
\begin{bmatrix}
\frac{1}{R_1} & -\frac{1}{R_1} & 0 & 1 & 0 \\
-\frac{1}{R_1} & \frac{1}{R_1} + \frac{1}{R_2} + \frac{1}{R_3} & -\frac{1}{R_2} & 0 & 0 \\
0 & -\frac{1}{R_2} & \frac{1}{R_2} & 0 & 1 \\
1 & 0 & 0 & 0 & 0 \\
0 & 0 & 1 & 0 & 0
\end{bmatrix}
\begin{bmatrix} v_a \\ v_b \\ v_c \\ i_{v1} \\ i_{v2} \end{bmatrix}
=
\begin{bmatrix} 0 \\ 0 \\ 0 \\ V1 \\ V2 \end{bmatrix}
$$

Now all that is left is to solve the 5x5 set of equations (recall that the nodal analysis method resulted in just 1 equation, though in that case we did some substitutions along the way). Solving the 5x5 equation is difficult by hand, but not so with a computer.

Example 2

If you'll recall, the nodal analysis method became a bit more difficult when one or more of the voltage sources was not connect to ground. Let's repeat Example 2 of the previous page with MNA. Here the circuit is repeated with steps 1 and 2 completed:



Steps 3 and 4

$$
\begin{aligned}
\frac{v_a}{R_1} - i_{v1} &= 0 \\
i_{v1} + \frac{v_b}{R_3} + \frac{v_b - v_c}{R_2} &= 0 \\
i_{v2} + \frac{v_c - v_b}{} &= 0
\end{aligned}
$$

$$i_{v2} + \frac{\quad}{R_2} = 0$$

$$v_b - v_a = V_1$$

$$v_c = V_2$$

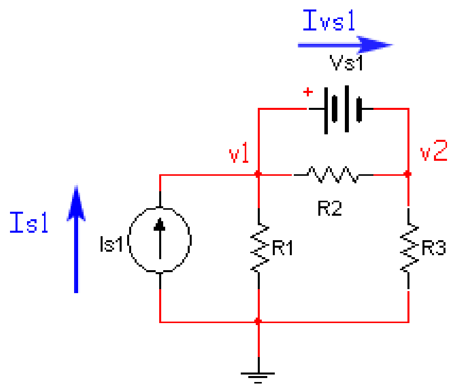
Step 5

$$\begin{bmatrix} \frac{1}{R_1} & 0 & 0 & -1 & 0 \\ 0 & \frac{1}{R_2}+\frac{1}{R_3} & -\frac{1}{R_2} & 1 & 0 \\ 0 & -\frac{1}{R_2} & \frac{1}{R_2} & 0 & 1 \\ -1 & 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 \end{bmatrix} \begin{bmatrix} v_a \\ v_b \\ v_c \\ i_{v1} \\ i_{v2} \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \\ 0 \\ V1 \\ V2 \end{bmatrix}$$

The fact that V_1 is not grounded presented no difficulty at all.

Example 3

Let's consider one more example, this time with a current source (this example is from Litovski). Steps 1 and 2 have been completed.



Now complete steps 3 and 4:

$$-I_{s1} + \frac{v_1}{R_1} + I_{Vs1} + \frac{v_1 - v_2}{R_2} = 0 \qquad (node\ 1)$$

$$-I_{Vs1} + \frac{v_2 - v_1}{R_2} + \frac{v_2}{R_3} = 0 \qquad (node\ 2)$$

$$v_1 - v_2 = V_{s1} \qquad (V_{s1})$$

And finally bring all the know variables to the right hand side and complete step 5:

$$\begin{bmatrix} \frac{1}{R_1}+\frac{1}{R_2} & -\frac{1}{R_2} & 1 \\ -\frac{1}{R_2} & \frac{1}{R_2}+\frac{1}{R_3} & -1 \\ 1 & -1 & 0 \end{bmatrix} \begin{bmatrix} v_1 \\ v_2 \\ I_{Vs1} \end{bmatrix} = \begin{bmatrix} I_{s1} \\ 0 \\ V_{s1} \end{bmatrix}$$

Observations about MNA

If you examine the matrix equations that resulted from the application of the MNA method, several patterns become apparent that we can use to develop an algorithm. All of the circuits resulted in an equation of the form.

$$\mathbf{A}\mathbf{x} = \mathbf{z}$$

Let us examine Example 2. This circuit had 3 nodes and 2 voltage sources (*n*=3, *m*=2). The resulting matrix is shown below.

$$\begin{bmatrix} \frac{1}{R_1} & 0 & 0 & -1 & 0 \\ 0 & \frac{1}{R_2}+\frac{1}{R_3} & -\frac{1}{R_2} & 1 & 0 \\ 0 & -\frac{1}{R_2} & \frac{1}{R_2} & 0 & 1 \\ -1 & 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 \end{bmatrix} \begin{bmatrix} v_a \\ v_b \\ v_c \\ i_{v1} \\ i_{v2} \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \\ 0 \\ V1 \\ V2 \end{bmatrix}$$

Note that the pink highlighted portion of the **A** matrix is 3×3 (in general *n*×*n*), and includes only known quantities, specifically the values of the passive elements (the resistors). In addition the highlighted portion of the **A** matrix is symmetric with positive values along the main diagonal, and only negative (or zero) values for the off-diagonal terms. If an element is connected to ground, it only appears along the diagonal; a non-grounded (e.g. R2) appears both on and off the diagonal). The rest of the terms in the **A** matrix (the non-highlighted portion) contains only ones, negative ones and zeros. Note also that the matrix size is 5x5 (in general (*m*+*n*)×(*m*+*n*)). For all of the circuits we will analyze (i.e., only passive elements and independent sources), these general observations about the **A** matrix will always hold.

Now consider the **x** matrix, the matrix of unknown quantities. It is a 5×1 matrix (in general (*n*+*m*)×1). The topmost 3 (in general *n*) elements are simply the node voltages. The bottom 2 (in general *m*) elements are the currents associated with the voltage sources.

This brings us to the **z** matrix that contains only known quantities. It is also a 5×1 matrix (in general (*n*+*m*)×1). The topmost 3 (in general *n*) elements are either zero, or the sum of independent current sources (see example 3 for an case in point). The bottom 2 (in general *m*) elements are the independent voltage sources.

To summarize:

MNA applied to a circuit with only passive elements (resistors) and independent current and voltage sources results in a matrix equation of the form:

$$\mathbf{A}\mathbf{x} = \mathbf{z}$$

For a circuit with *n* nodes and *m* independent voltage sources:

- The **A** matrix:

- The **A** matrix:
    - is $(n+m)\times(n+m)$ in size, and consists only of known quantities.
    - the $n\times n$ part of the matrix in the upper left:
        - has only passive elements
        - elements connected to ground appear only on the diagonal
        - elements not connected to ground are both on the diagonal and off-diagonal terms.
    - the rest of the **A** matrix (not included in the $n\times n$ upper left part) contains only 1, -1 and 0 (other values are possible if there are dependent current and voltage sources; I have not considered these cases. Consult Litovski if interested.)
- The **x** matrix:
    - is an $(n+m)\times1$ vector that holds the unknown quantities (node voltages and the currents through the independent voltage sources).
    - the top n elements are the n node voltages.
    - the bottom m elements represent the currents through the m independent voltage sources in the circuit.
- The **z** matrix:
    - is an $(n+m)\times1$ vector that holds only known quantities
    - the top n elements are either zero or the sum and difference of independent current sources in the circuit.
    - the bottom m elements represent the m independent voltage sources in the circuit

An Algorithm for Modified Nodal Analysis

This document describes an algorithmic method for generating MNA (Modified Nodal Analysis) equations for systems with only resistors and idependent sources.. It consists of several parts:.

Contents

Many of the ideas and notation from this page are from Litovski, though the discussion here is quite simpler because only independent voltage and current sources are considered.

Review

Recall from the previous document:

MNA applied to a circuit with only passive elements (resistors) and independent current and voltage sources results in a matrix equation of the form:

$$\mathbf{Ax} = \mathbf{z}$$

For a circuit with *n* nodes and *m* independent voltage sources:

- The **A** matrix:
  - is (*n*+*m*)x(*n*+*m*) in size, and consists only of known quantities.
  - the *n*x*n* part of the matrix in the upper left:
    - has only passive elements
    - elements connected to ground appear only on the diagonal
    - elements not connected to ground are both on the diagonal and off-diagonal terms.
  - the rest of the **A** matrix (not included in the *n*x*n* upper left part)

contains only 1, -1 and 0 (other values are possible if there are dependent current and voltage sources; I have not considered these cases. Consult Litovski if interested.)

- The **x** matrix:
  - is an (n+m)x1 vector that holds the unknown quantities (node voltages and the currents through the independent voltage sources).
  - the top *n* elements are the *n* node voltages.
  - the bottom *m* elements represent the currents through the *m* independent voltage sources in the circuit.
- The **z** matrix:
  - is an (n+m)x1 vector that holds only known quantities
  - the top *n* elements are either zero or the sum and difference of independent current sources in the circuit.
  - the bottom *m* elements represent the *m* independent voltage sources in the circuit.

The circuit is solved by a simple matrix manipulation:

$$\mathbf{x} = \mathbf{A}^{-1}\mathbf{z}$$

Though this may be difficult by hand, it is straightforward and so is easily done by computer.

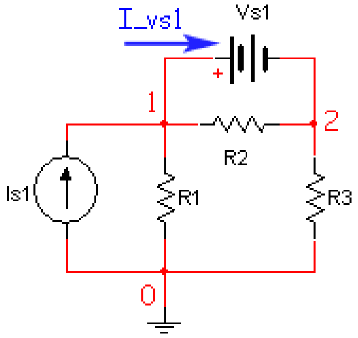
Notational Convention

Obviously, the notation used does not change the solution. However the convention described below will make it quite easy to develop the matrices necessary for solution of the circuit.

- Ground is labeled as node 0.
- The other nodes are labeled consecutively from 1 to *n*.
- We will refer to the voltage at node 1 as v_1, at node 2 as v_2 and so on.
- The naming of the independent voltage sources is quite loose, but the names must start with the letter "V" and must be unique from any node names. For our purposes we will require that independent voltage sources have no underscore ("_") in their names. So the names Va, Vsource, V1, Vxyz123 are all legitimate names, but V_3, V_A, Vsource_1 are not.
- The current through a voltage source will be labeled with "I_" followed by the name of the voltage source. Therefore the current through Va is I_Va, the current through VSource is I_VSource, etc...
- The naming of the independent current sources is similar; the names must start with the letter "I" and must have no underscore ("_") in their names. So the names Ia, Isource, I1, Ixyz123 are all legitimate names, but I_3, I_A, Isource_1 are not.
- *Note: some of these rules are arbitrary (particularly the naming conventions) but will make development of a MATLAB based algorithm easier.*

These rules are somewhat restrictive (more so than they need to be) but they make development of the algorithm easier while still allowing quite a bit of make development of the algorithm easier while still allowing quite a bit of freedom.

These rules are easily explained with an example (Example 3 from previous page):



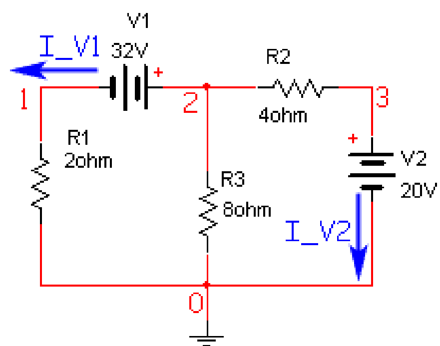
This circuit is labeled according to the guidelines above. Ground is node 0 and the other two nodes are labeled 1 and 2. The voltage and current sources have labels with no underscores and the current through the voltage source Vs1 is labeled I_Vs1. We will discuss how to use this diagram (with these labels) to generate the MNA equations below.

Generating the MNA matrices

There are three matrices we need to generate, the **A** matrix, the **x** matrix and the **z** matrix. Each of these will be created by combining several individual sub-matrices. To motivate the rules for generating the matrices we will consider the two sample circuits below (example 2 and example 3 from previous web page, with Notational Convention as above). I simply state the resulting MNA equations, but in the following text we will show how to generate each one.

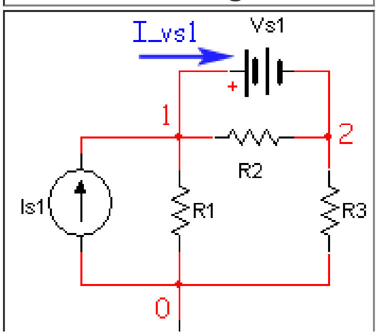
Case 1

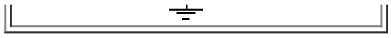
Circuit Diagram



Case 2

Circuit Diagram





The A matrix

The **A** matrix will be developed as the combination of 4 smaller matrices, **G**, **B**, **C**, and **D**.

$$\mathbf{A} = \begin{bmatrix} \mathbf{G} & \mathbf{B} \\ \mathbf{C} & \mathbf{D} \end{bmatrix}$$

The **A** matrix is $(m+n)\times(m+n)$ (n is the number of nodes, and m is the number of independent voltage sources) and:

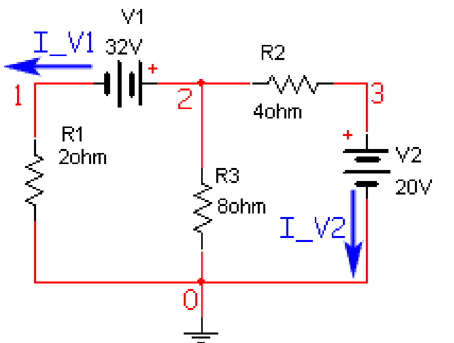
- the **G** matrix is $n\times n$ and is determined by the interconnections between the passive circuit elements (resistors)
- the **B** matrix is $n\times m$ and is determined by the connection of the voltage sources.
- the **C** matrix is $m\times n$ and is determined by the connection of the voltage sources. (**B** and **C** are closely related, particularly when only independent sources are considered).
- the **D** matrix is $m\times m$ and is zero if only independent sources are considered.

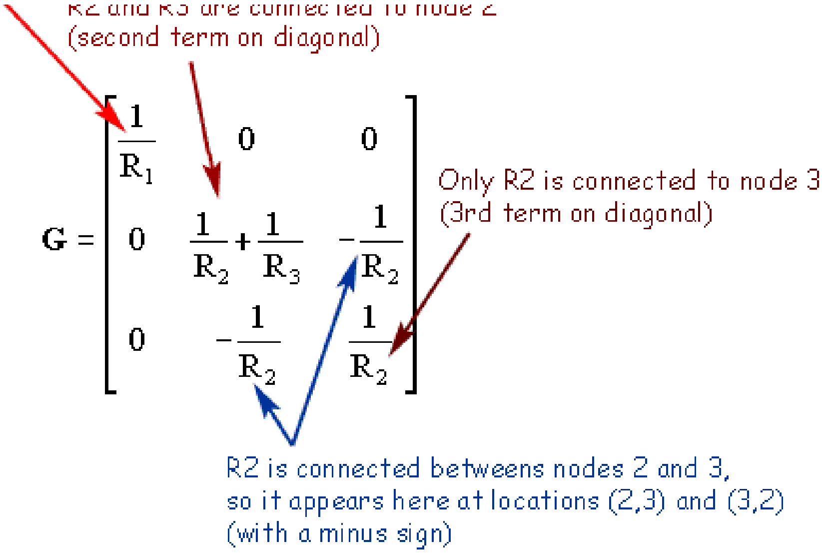
Rules for making the G matrix

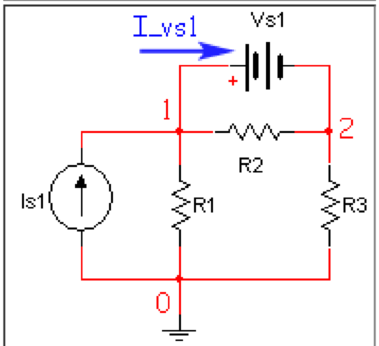
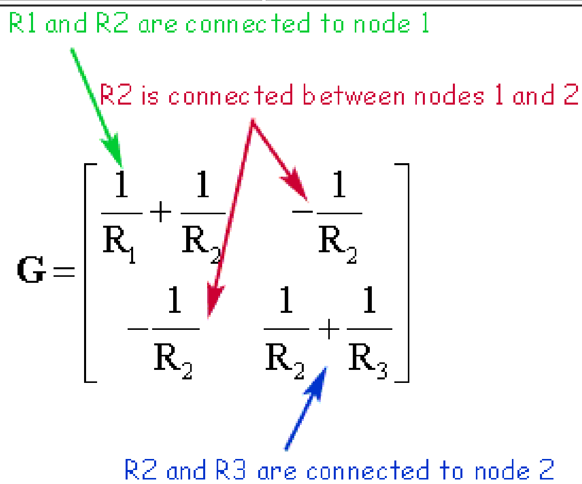
The **G** matrix is an $n\times n$ matrix formed in two steps

1. Each element in the diagonal matrix is equal to the sum of the conductance (one over the resistance) of each element connected to the corresponding node. So the first diagonal element is the sum of conductances connected to node 1, the second diagonal element is the sum of conductances connected to node 2, and so on.
2. The off diagonal elements are the negative conductance of the element connected to the pair of corresponding node. Therefore a resistor between nodes 1 and 2 goes into the **G** matrix at location (1,2) and locations (2,1).

Demonstrations

Case 1	
Circuit Diagram	MNA Equations
	$\mathbf{G} = \begin{bmatrix} \frac{1}{R_1} & 0 & 0 \\ 0 & \frac{1}{R_2}+\frac{1}{R_3} & -\frac{1}{R_2} \\ 0 & -\frac{1}{R_2} & \frac{1}{R_2} \end{bmatrix}$
R1 is connected to node 1 (First term on diagonal) R2 and R3 are connected to node 2	



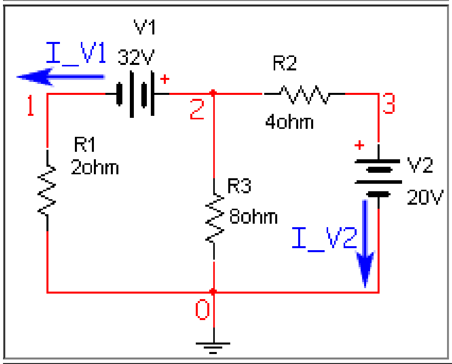
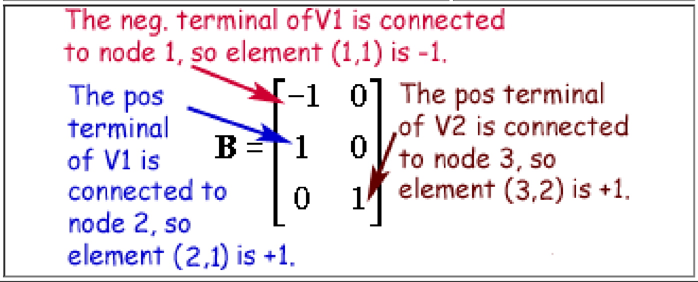
Case 2	
Circuit Diagram	MNA Equations
	$\mathbf{G} = \begin{bmatrix} \frac{1}{R_1} + \frac{1}{R_2} & -\frac{1}{R_2} \\ -\frac{1}{R_2} & \frac{1}{R_2} + \frac{1}{R_3} \end{bmatrix}$
	

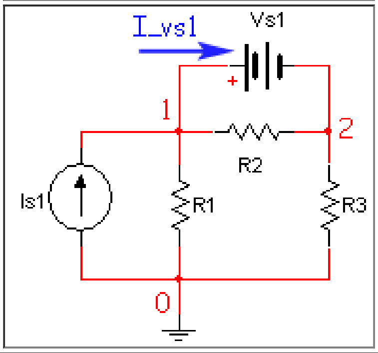
If an element is grounded, it will only have contribute to one entry in the **G** matrix -- at the appropriate location on the diagonal. If it is ungrounded it will contribute to four entries in the matrix -- two diagonal entries (corresponding to the two nodes) and two off-diagonal entries.

Rules for making the B matrix

The **B** matrix is an $n \times m$ matrix with only 0, 1 and -1 elements. Each location in the matrix corresponds to a particular voltage source (first dimension) or a node (second dimension). If the positive terminal of the *i*th voltage source is connected (second dimension). If the positive terminal of the *i*th voltage source is connected to node *k*, then the element (*i*,*k*) in the **B** matrix is a 1. If the negative terminal of the *i*th voltage source is connected to node *k*, then the element (*i*,*k*) in the **B** matrix is a -1. Otherwise, elements of the **B** matrix are zero.

Demonstrations

Case 1	
Circuit Diagram	MNA Equations
	$\mathbf{B}=\begin{bmatrix}-1 & 0\\ 1 & 0\\ 0 & 1\end{bmatrix}$
	

Case 2	
Circuit Diagram	MNA Equations
	$\mathbf{B}=\begin{bmatrix}1\\ -1\end{bmatrix}$

If a voltage source is ungrounded, it will have two elements in the **B** matrix (a 1 and a -1 in the same column). If it is grounded it will only have one element in the matrix.

Rules for making the C matrix

The **C** matrix is an *m*×*n* matrix with only 0, 1 and -1 elements. Each location in the matrix corresponds to a particular node (first dimension) or voltage source (second dimension). If the positive terminal of the *i*th voltage source is connected to node *k*, then the element (*k*,*i*) in the **C** matrix is a 1. If the negative terminal of the *i*th voltage source is connected to node *k*, then the element (*k*,*i*) in the **C** matrix is a -1. Otherwise, elements of the **C** matrix are zero.

In other words, the **C** matrix is the transpose of the **B** matrix. (This is not the case

In other words, the C matrix is the transpose of the B matrix. (This is not the case when dependent sources are present.)

Case 1: $\mathbf{C}=\begin{bmatrix}-1 & 1 & 0\\ 0 & 0 & 1\end{bmatrix}$

Case 2: $\mathbf{C}=\begin{bmatrix}1 & -1\end{bmatrix}$

Rules for making the D matrix

The **D** matrix is an *m*×*m* matrix that is composed entirely of zeros. (It can be non-zero if dependent sources are considered.)

Case 1: $\mathbf{D}=\begin{bmatrix}0 & 0\\ 0 & 0\end{bmatrix}$

Case 2: $\mathbf{D}=\begin{bmatrix}0\end{bmatrix}$

The x matrix

The **x** matrix holds our unknown quantities and will be developed as the combination of 2 smaller matrices **v** and **j**. It is considerably easier to define than the **A** matrix.

$$\mathbf{x}=\begin{bmatrix}\mathbf{v}\\ \mathbf{j}\end{bmatrix}$$

The **x** matrix is (*m*+*n*)×1 (*n* is the number of nodes, and *m* is the number of independent voltage sources) and:

- the **v** matrix is *n*×1 and hold the unknown voltages
- the **j** matrix is *m*×1 and holds the unknown currents through the voltage sources

Rules for making the v matrix

The **v** matrix is an *n*×1 matrix formed of the node voltages. Each element in **v** corresponds to the voltage at the equivalent node in the circuit (there is no entry for ground -- node 0).

For example if a circuit has three nodes, the **v** matrix is

$$\mathbf{v}=\begin{bmatrix}v\_1\\ v\_2\\ v\_3\end{bmatrix}$$

For a circuit with *n* nodes we get

$$\mathbf{v}==\begin{bmatrix}v\_1\\ \vdots\\ v\_n\end{bmatrix}$$

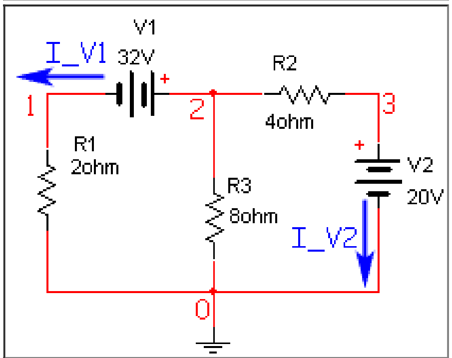
and so on.

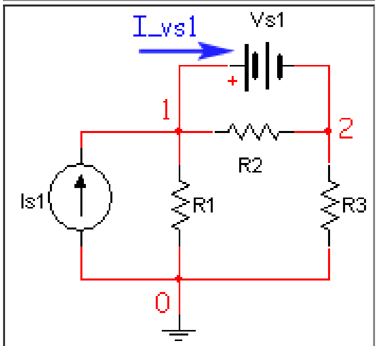
Rules for making the j matrix

The **j** matrix is an *m*×1 matrix, with one entry for the current through each voltage source. So if there are two voltage sources V1 and V2, the **j** matrix will be:

$$\mathbf{j} = \begin{bmatrix} i\_V1 \\ i\_V2 \end{bmatrix}$$

Demonstrations for the **x** *matrix.*

Case 1	
Circuit Diagram	MNA Equations
	$\mathbf{v} = \begin{bmatrix} v\_1 \\ v\_2 \\ v\_3 \end{bmatrix}, \mathbf{j} = \begin{bmatrix} i\_V1 \\ i\_V2 \end{bmatrix}, \quad \mathbf{x} = \begin{bmatrix} \mathbf{v} \\ \mathbf{j} \end{bmatrix} = \begin{bmatrix} v\_1 \\ v\_2 \\ v\_3 \\ i\_V1 \\ i\_V2 \end{bmatrix}$

Case 2	
Circuit Diagram	MNA Equations
	$\mathbf{v} = \begin{bmatrix} v\_1 \\ v\_2 \end{bmatrix}, \mathbf{j} = \begin{bmatrix} i\_Vs1 \end{bmatrix}, \quad \mathbf{x} = \begin{bmatrix} \mathbf{v} \\ \mathbf{j} \end{bmatrix} = \begin{bmatrix} v\_1 \\ v\_2 \\ i\_Vs1 \end{bmatrix}$

The z matrix

The **z** matrix holds our independent voltage and current sources and will be developed as the combination of 2 smaller matrices **i** and **e**. It is quite easy to formulate.

$$\mathbf{z} = \begin{bmatrix} \mathbf{i} \\ \mathbf{e} \end{bmatrix}$$

The **z** matrix is (*m*+*n*)×1 (*n* is the number of nodes, and *m* is the number of independent voltage sources) and:

- the **i** matrix is *n*×1 and contains the sum of the currents through the passive elements into the corresponding node (either zero, or the sum of independent current sources).

- the **e** matrix is m×1 and holds the values of the independent voltage sources.

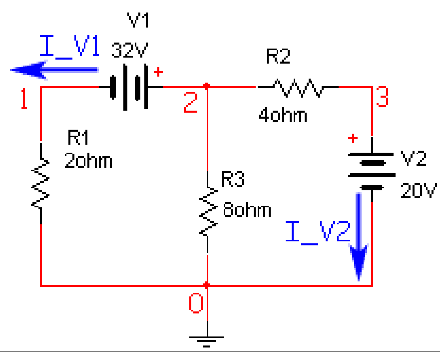
Rules for making the i matrix

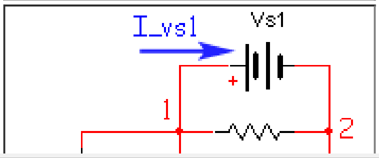
The **i** matrix is an n×1 matrix with each element of the matrix corresponding to a particular node. The value of each element of **i** is determined by the sum of current sources into the corresponding node. If there are no current sources connected to the node, the value is zero.

Rules for making the e matrix

The **e** matrix is an m×1 matrix with each element of the matrix equal in value to the corresponding independent voltage source.

Demonstrations for the ***z*** *matrix.*

Case 1	
Circuit Diagram	MNA Equations
	$\mathbf{i}=\begin{bmatrix}0\\0\\0\end{bmatrix},\ \mathbf{e}=\begin{bmatrix}V1\\V2\end{bmatrix},\ \mathbf{z}=\begin{bmatrix}0\\0\\0\\V1\\V2\end{bmatrix}$

Case 2	
Circuit Diagram	MNA Equations
	

Some Examples of Modified Nodal Analysis

Introduction | MNA Basics | Algorithmic MNA | Examples | Reacitve Circuits and OpAmps | SCAM | Dependent Sources | Printable

This document describes an algorithmic method for generating MNA (Modified Nodal Analysis) equations for systems with only impedances (resistors) and independent voltage and current sources. It consists of several parts:.

Contents

Many of the ideas and notation from this page are from Litovski, though the discussion here is quite simpler because only passive elements (*e.g.*, resistors) independent voltage and current sources are considered.

Review

Recall from the previous document that MNA applied to a circuit results in a matrix equation of the form:

$$\mathbf{Ax} = \mathbf{z}$$

We will take *n* to be the number of nodes (not including ground) and *m* to be the number of independent voltage sources.

Notation

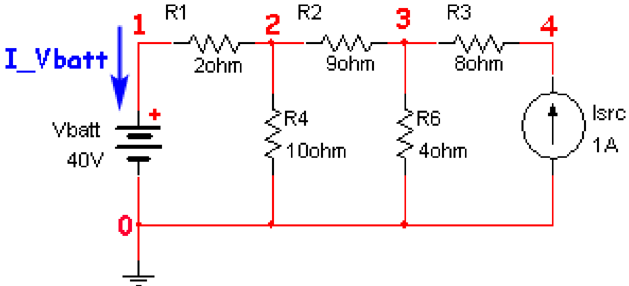
- Ground is labeled as node 0.
- The other nodes are labeled consecutively from 1 to *n*.
- We will refer to the voltage at node 1 as v_1, at node 2 as v_2 and so on.
- The naming of the independent voltage sources is quite loose, but the names must start with the letter "V" and must be unique from any node names. For our purposes we will require that independent voltage sources have no underscore ("_") in their names. So the names Va, Vsource, V1, Vxyz123 are all legitimate names, but V_3, V_A, Vsource_1 are not.
- The current through a voltage source will be labeled with "I_" followed by the name of the voltage source. Therefore the current through Va is I_Va, the current through VSource is I_VSource, etc...
- The naming of the independent current sources is similar; the names must start with the letter "I" and must have no underscore ("_") in their names. So the names Ia, Isource, I1, Ixyz123 are all legitimate names, but I_3, I_A, Isource_1 are not.
- *Note: some of these rules are arbitrary (particularly the naming conventions) but will make development of a MATLAB based algorithm easier.*

development of a MATLAB based algorithm easier.

Review the rules for forming MNA matrices (if needed).

Example 1

The example given is from Smith, Figure 2.8. First the MNA equations will be derived from the circuit. They will then be derived according to the algorithm in the previous document -- hopefully the results will agree. The nodes and sources have been labeled as required, and the current through the voltage source is defined (as required).



To apply the MNA technique we will need 5 equations (one for each of 4 nodes, and 1 for the independent voltage source). Note that Smith only required 2 equations -- MNA often requires more equations than other techniques, but is amenable to computer solution. By inspection we get:

$$\begin{aligned}
I_Vbatt + \frac{v_1 - v_2}{R1} &= 0 && node\ 1 \\
\frac{v_2 - v_1}{R1} + \frac{v_2}{R4} + \frac{v_3 - v_2}{R2} &= 0 && node\ 2 \\
\frac{v_2 - v_3}{R2} + \frac{v_3}{R6} + \frac{v_4 - v_3}{R3} &= 0 && node\ 3 \\
\frac{v_3 - v_4}{R3} - Isrc &= 0 && node\ 4 \\
v_1 &= Vbatt && voltage\ source
\end{aligned}$$

Using the algorithm, we get:

$$\mathbf{G} = \begin{bmatrix} \frac{1}{R1} & -\frac{1}{R1} & 0 & 0 \\ -\frac{1}{R1} & \frac{1}{R1}+\frac{1}{R4}+\frac{1}{R2} & -\frac{1}{R2} & 0 \\ 0 & -\frac{1}{R2} & \frac{1}{R2}+\frac{1}{R6}+\frac{1}{R3} & -\frac{1}{R3} \\ 0 & 0 & -\frac{1}{R3} & \frac{1}{R3} \end{bmatrix}, \mathbf{B} = \mathbf{C}^T = \begin{bmatrix} 1 \\ 0 \\ 0 \\ 0 \end{bmatrix}, \mathbf{D} =$$

$$\mathbf{A} = \begin{bmatrix} \frac{1}{R1} & -\frac{1}{R1} & 0 & 0 & 1 \\ -\frac{1}{R1} & \frac{1}{R1}+\frac{1}{R4}+\frac{1}{R2} & -\frac{1}{R2} & 0 & 0 \\ 0 & -\frac{1}{R2} & \frac{1}{R2}+\frac{1}{R6}+\frac{1}{R3} & -\frac{1}{R3} & 0 \\ 0 & 0 & -\frac{1}{R3} & \frac{1}{R3} & 0 \\ 1 & 0 & 0 & 0 & 0 \end{bmatrix}$$

$$\mathbf{v} = \begin{bmatrix} v\_2 \\ v\_3 \\ v\_4 \end{bmatrix}, \quad \mathbf{j} = [I\_Vbatt], \quad \mathbf{x} = \begin{bmatrix} v\_2 \\ v\_3 \\ v\_4 \\ I\_Vbatt \end{bmatrix}$$

$$\mathbf{i} = \begin{bmatrix} 0 \\ 0 \\ 0 \\ Isrc \end{bmatrix}, \quad \mathbf{e} = [Vbatt], \quad \mathbf{z} = \begin{bmatrix} 0 \\ 0 \\ 0 \\ Isrc \\ Vbatt \end{bmatrix}$$

Putting these together yields:

$$\mathbf{Ax} = \mathbf{z}$$

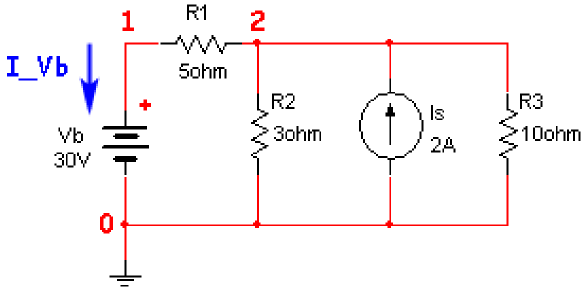
$$or$$

$$\begin{bmatrix} \frac{1}{R1} & -\frac{1}{R1} & 0 & 0 & 1 \\ -\frac{1}{R1} & \frac{1}{R1}+\frac{1}{R4}+\frac{1}{R2} & -\frac{1}{R2} & 0 & 0 \\ 0 & -\frac{1}{R2} & \frac{1}{R2}+\frac{1}{R6}+\frac{1}{R3} & -\frac{1}{R3} & 0 \\ 0 & 0 & -\frac{1}{R3} & \frac{1}{R3} & 0 \\ 1 & 0 & 0 & 0 & 0 \end{bmatrix} \begin{bmatrix} v\_1 \\ v\_2 \\ v\_3 \\ v\_4 \\ I\_Vbatt \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \\ 0 \\ Isrc \\ Vbatt \end{bmatrix}$$

Careful comparison of this result with the original result verifies that the two solutions are identical.

Example 2

The example given is from Smith, Problem 2.11. First the MNA equations will be derived from the circuit. They will then be derived according to the algorithm in the previous document -- hopefully the results will agree. The nodes and sources have been labeled as required, and the current through the voltage source is defined.



To apply the MNA technique we will need 3 equations (one for each of 2 nodes, and 1 for the independent voltage source). By inspection we get:

$$I\_Vb + \frac{v\_1 - v\_2}{R1} = 0 \qquad node\ 1$$

$$\frac{v\_2 - v\_1}{R1} + \frac{v\_2}{R2} + \frac{v\_2}{R3} - Is = 0 \qquad node\ 2$$

$$v\_1 = Vb \qquad voltage\ source$$

Using the algorithm, we get:

$$\mathbf{G} = \begin{bmatrix} \frac{1}{R1} & -\frac{1}{R1} \\ -\frac{1}{R1} & \frac{1}{R1} + \frac{1}{R2} + \frac{1}{R3} \end{bmatrix}, \quad \mathbf{B} = \mathbf{C}^T = \begin{bmatrix} 1 \\ 0 \end{bmatrix}, \quad \mathbf{D} = [0]$$

$$\mathbf{A} = \begin{bmatrix} \frac{1}{R1} & -\frac{1}{R1} & 1 \\ -\frac{1}{R1} & \frac{1}{R1} + \frac{1}{R2} + \frac{1}{R3} & 0 \\ 1 & 0 & 0 \end{bmatrix}$$

$$\mathbf{v} = \begin{bmatrix} v\_1 \\ v\_2 \end{bmatrix}, \quad \mathbf{j} = [I\_Vb], \quad \mathbf{x} = \begin{bmatrix} v\_1 \\ v\_2 \\ I\_Vb \end{bmatrix}$$

$$\mathbf{i} = \begin{bmatrix} 0 \\ Is \end{bmatrix}, \quad \mathbf{e} = [Vb], \quad \mathbf{z} = \begin{bmatrix} 0 \\ Is \\ Vb \end{bmatrix}$$

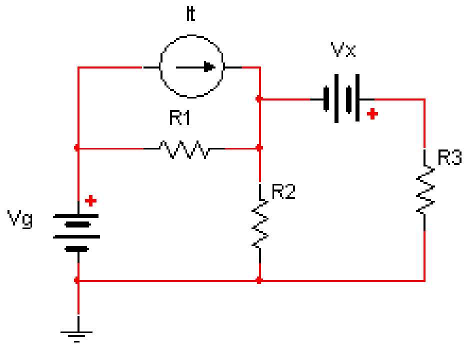
If we apply these results to the MNA equation, we get

$$\mathbf{A}\mathbf{x} = \mathbf{z}$$
$$or$$
$$\begin{bmatrix} \frac{1}{R1} & -\frac{1}{R1} & 1 \\ -\frac{1}{R1} & \frac{1}{R1} + \frac{1}{R2} + \frac{1}{R3} & 0 \\ 1 & 0 & 0 \end{bmatrix} \begin{bmatrix} v\_1 \\ v\_2 \\ I\_Vb \end{bmatrix} = \begin{bmatrix} 0 \\ Is \\ Vb \end{bmatrix}$$

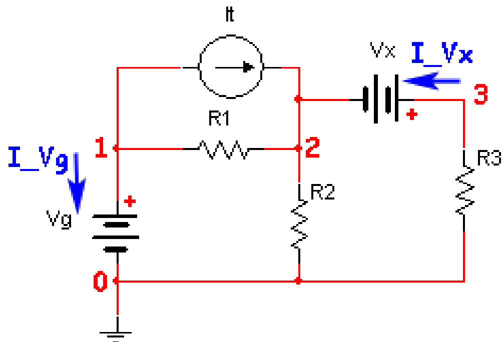
Careful comparison of this result with the original result verifies that the two solutions are identical.

Example 3

The last example is a bit more involved, it has two voltage sources and one current source. The current source and one of the voltage sources are not grounded.



First we must label the nodes and define currents through the voltage sources



To apply the MNA technique we will need 5 equations (one for each of 3 nodes, and

To apply the MNA technique we will need 5 equations (one for each of 3 nodes, and 2 for the independent voltage sources). By inspection we get:

$$\begin{aligned} I\_Vg + It + \frac{v\_1 - v\_2}{R1} &= 0 && node\ 1 \\ -It + \frac{v\_2 - v\_1}{R1} + \frac{v\_2}{R2} - I\_Vx &= 0 && node\ 2 \\ I\_Vx + \frac{v\_3}{R3} &= 0 && node\ 2 \\ v\_1 &= Vg && voltage\ source\ 1 \\ v\_3 - v\_2 &= Vx && voltage\ source\ 2 \end{aligned}$$

Using the algorithm we get:

$$\mathbf{G} = \begin{bmatrix} \frac{1}{R1} & -\frac{1}{R1} & 0 \\ -\frac{1}{R1} & \frac{1}{R1} + \frac{1}{R2} & 0 \\ 0 & 0 & \frac{1}{R3} \end{bmatrix}, \mathbf{B} = \mathbf{C}^T = \begin{bmatrix} 1 & 0 \\ 0 & -1 \\ 0 & 1 \end{bmatrix}, \mathbf{D} = \begin{bmatrix} 0 & 0 \\ 0 & 0 \end{bmatrix}$$

$$\mathbf{A} = \begin{bmatrix} \frac{1}{R1} & -\frac{1}{R1} & 0 & 1 & 0 \\ -\frac{1}{R1} & \frac{1}{R1} + \frac{1}{R2} & 0 & 0 & -1 \\ 0 & 0 & \frac{1}{R3} & 0 & 1 \\ 1 & 0 & 0 & 0 & 0 \\ 0 & -1 & 1 & 0 & 0 \end{bmatrix}$$

$$\mathbf{v} = \begin{bmatrix} v\_1 \\ v\_2 \\ v\_3 \end{bmatrix}, \mathbf{j} = \begin{bmatrix} I\_Vg \\ I\_Vx \end{bmatrix}, \quad \mathbf{x} = \begin{bmatrix} v\_1 \\ v\_2 \\ v\_3 \\ I\_Vg \\ I\_Vx \end{bmatrix}$$

$$\mathbf{i} = \begin{bmatrix} -It \\ It \\ 0 \end{bmatrix}, \mathbf{e} = \begin{bmatrix} Vg \\ Vx \end{bmatrix}, \quad \mathbf{z} = \begin{bmatrix} -It \\ It \\ 0 \\ Vg \\ Vx \end{bmatrix}$$

If we apply these results to the MNA equation, we get

$$\mathbf{Ax} = \mathbf{z}$$
$$or$$
$$\begin{bmatrix} \frac{1}{R1} & -\frac{1}{R1} & 0 & 1 & 0 \\ -\frac{1}{R1} & \frac{1}{R1} + \frac{1}{R2} & 0 & 0 & -1 \\ 0 & 0 & \frac{1}{R3} & 0 & 1 \\ 1 & 0 & 0 & 0 & 0 \\ 0 & -1 & 1 & 0 & 0 \end{bmatrix} \begin{bmatrix} v\_1 \\ v\_2 \\ v\_3 \\ I\_Vg \\ I\_Vx \end{bmatrix} = \begin{bmatrix} -It \\ It \\ 0 \\ Vg \\ Vx \end{bmatrix}$$

Careful comparison of this result with the original result verifies that the two solutions are identical.

References

Modified Nodal Analysis with Inductors, Capacitors and Op-Amps

This document describes an algorithmic method for generating modified nodal analysis (MNA) equations when the circuit has inductors, capacitors and/or operational amplifiers (op-amps). It consists of several parts:.

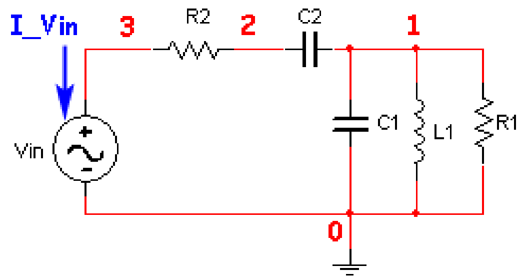
Contents

MNA with Reactive Elements

Applying modified nodal analysis to circuits with inductors and capacitors presents no special difficulty if one uses the complex impedance of these elements.

$$Z_R = R$$
$$Z_L = sL = j\omega L$$
$$Z_C = \frac{1}{sC} = \frac{1}{j\omega C}$$

Let us apply MNA to the following circuit (which already has nodes labeled, and the current through the voltage source defined and labeled):



MNA will generate 4 equations, one for each of the three nodes, and one for Vin. By inspection we get:

$$v\_1 \cdot s \cdot C1 + \frac{v\_1}{s \cdot L1} + \frac{v\_1}{R1} + (v\_1 - v\_2) \cdot s \cdot C2 = 0 \qquad node\ 1$$
$$\frac{v\_2 - v\_3}{R2} + (v\_2 - v\_1) \cdot s \cdot C2 = 0 \qquad node\ 2$$
$$I\_Vin + \frac{v\_3 - v\_2}{R2} = 0 \qquad node\ 3$$
$$v\_3 = Vin \qquad Vin$$

Using the MNA algorithm we get:

$$\mathbf{G} = \begin{bmatrix} s \cdot C1 + s \cdot C2 + \frac{1}{s \cdot L1} + \frac{1}{R1} & -s \cdot C2 & 0 \\ -s \cdot C2 & s \cdot C2 + \frac{1}{R2} & -\frac{1}{R2} \\ 0 & -\frac{1}{R2} & \frac{1}{R2} \end{bmatrix}, \qquad \mathbf{B} = \mathbf{C}^T = \begin{bmatrix} 0 \\ 0 \\ 1 \end{bmatrix}$$

$$\mathbf{A} = \begin{bmatrix} s \cdot C1 + s \cdot C2 + \frac{1}{s \cdot L1} + \frac{1}{R1} & -s \cdot C2 & 0 & 0 \\ -s \cdot C2 & s \cdot C2 + \frac{1}{R2} & -\frac{1}{R2} & 0 \\ 0 & -\frac{1}{R2} & \frac{1}{R2} & 1 \\ 0 & 0 & 1 & 0 \end{bmatrix}$$

$$\mathbf{v} = \begin{bmatrix} v\_1 \\ v\_2 \\ v\_3 \end{bmatrix}, \qquad \mathbf{j} = [I\_Vin], \qquad \mathbf{x} = \begin{bmatrix} v\_1 \\ v\_2 \\ v\_3 \\ I\_Vin \end{bmatrix}$$

$$\mathbf{i} = \begin{bmatrix} 0 \\ 0 \\ 0 \end{bmatrix}, \qquad \mathbf{e} = [Vin], \qquad \mathbf{z} = \begin{bmatrix} 0 \\ 0 \\ 0 \\ Vin \end{bmatrix}$$

Since the matrices are all defined, we can now wirte the final matrices:

$$\mathbf{Ax} = \mathbf{z}$$

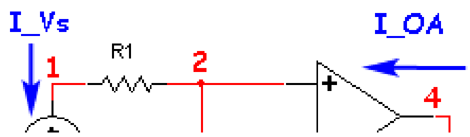
$$or$$

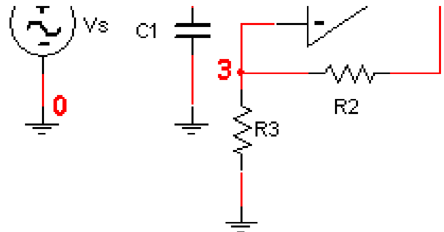
$$\begin{bmatrix} s \cdot C1 + s \cdot C2 + \frac{1}{s \cdot L1} + \frac{1}{R1} & -s \cdot C2 & 0 & 0 \\ -s \cdot C2 & s \cdot C2 + \frac{1}{R2} & -\frac{1}{R2} & 0 \\ 0 & -\frac{1}{R2} & \frac{1}{R2} & 1 \\ 0 & 0 & 1 & 0 \end{bmatrix} \begin{bmatrix} v\_1 \\ v\_2 \\ v\_3 \\ I\_Vin \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \\ 0 \\ Vin \end{bmatrix}$$

Careful inspection of this result verifies that it is identical to the original result. Don't worry about solving this equation; a later page will introduce SCAM - A MATLAB tool for deriving and solving circuit equations symbolically.

MNA with Op Amps

Applying modified nodal analysis to circuits with ideal operational amplifiers (op- amps) is a bit more difficult. Each op-amp increases the count of voltage sources by 1 (because the output of an op amp is treated as a voltage source), but also complicates the creation of the MNA matrices. In particular, the **B** and **C** matrices are no longer transposes of each other. To see how the ideal operational amplifier is handled, consider the circuit below:





Note that we have labelled all of the nodes, and defined a current through each voltage source. The current through the voltage source Vs is I_Vs, and the voltage into the op-amp is labeled I_OA. We will make the standard assumption for an ideal op-amp. Namely that there is no current into the device at either input to the op-amp, and the voltage difference between the inputs is zero (note caveats below).

This circuit will require 6 equations -- one each for the 4 nodes and one each for the 2 labeled currents. We can now write the circuit equations by inspection.

$$\begin{aligned}
I_Vs+\frac{v_1-v_2}{R1}&=0 && node\ 1\\
\frac{v_2-v_1}{R1}+v_2\cdot s\cdot C1&=0 && node\ 2\\
\frac{v_3}{R3}+\frac{v_3-v_4}{R2}&=0 && node\ 3\\
I_OA+\frac{v_4-v_3}{R2}&=0 && node\ 4\\
v_1&=Vs && Vs\\
v_2-v_3&=0 && OpAmp\ inputs
\end{aligned}$$

or in matrix form:

$$\begin{bmatrix}
\frac{1}{R1} & -\frac{1}{R1} & 0 & 0 & 1 & 0\\
-\frac{1}{R1} & \frac{1}{R1}+s\cdot C1 & 0 & 0 & 0 & 0\\
0 & 0 & \frac{1}{R2}+\frac{1}{R3} & -\frac{1}{R2} & 0 & 0\\
0 & 0 & -\frac{1}{R2} & \frac{1}{R2} & 0 & 1\\
1 & 0 & 0 & 0 & 0 & 0\\
0 & 1 & -1 & 0 & 0 & 0
\end{bmatrix}
\begin{bmatrix} v_1\\ v_2\\ v_3\\ v_4\\ I_Vs\\ I_OA \end{bmatrix}
=
\begin{bmatrix} 0\\ 0\\ 0\\ 0\\ Vs\\ 0 \end{bmatrix}$$

The only difference between this equation and the others that we have developed is that the equation for the op-amp is not in terms of the voltage at its output, but rather it specifies that the two input terminals are at the same potential.

Changes to formation of the MNA matrices.

The **A** matrix:

Recall that the **A** matrix is formed of four smaller matrices, **G**, **B**, **C**, and **D**.

- The rule for the **G** matrix is unchanged.
- The rule for the **B** matrix is unchanged -- the op-amp is treated as another

- The rule for the **B** matrix is unchanged – the op-amp is treated as another voltage source.
- The rule for the **C** matrix does change. The **C** matrix is an *n*×*m* matrix with only 0, 1 and -1 elements. Each location in the matrix corresponds to a particular node (first dimension) or voltage source (second dimension). For each indendent voltage source, if the positive terminal of the *i*th voltage source is connected to node *k*, then the element (*k,i*) in the **C** matrix is a 1; if the negative terminal of the *i*th voltage source is connected to node *k*, then the element (*k,i*) in the **C** matrix is a -1. For each op-amp let the positive input terminal be at node *k* and negative terminal at node *j*; the corresponding (*i*th) row of the C matrix has a 1 at location corresponding to the positive terminal (*k,i*), and a -1 at the location corresponding to the negative terminal (*j,i*). Otherwise, elements of the **C** matrix are zero.
- The rule for the **D** matrix is unchanged.

The **x** matrix is unchanged.

The **z** matrix:

Recall that the **z** matrix is formed of two smaller matrices, **i** and **e**.

- The rule for the **i** matrix is unchanges.
- The rule for the **e** matrix does change. The **e** matrix is an 1×*m* matrix with each element of the matrix corresponding to a voltage source. If the element in the **e** matrix corresponds to an independent source it is set equal to the value of that voltage source. If the element corresponds to an op-amp, then its value is set to zero.

Caveats: The two ideal op-amp assumptions (no input current, no potential difference at inputs) only hold for circuits with negative feedback. Because of

SCAM: Symbolic Circuit Analysis in MatLab

This document describes a MATLAB® tool for deriving and solving circuit equations symbolically. It is split up into several segments.

Contents

Downloading SCAM

Before using the program you must first download it from GitHub and save it on your computer where MATLAB can access it. Note: you must have the symbolic toolbox to run this code. The example netlists described on this page are also on GitHub.

About the code

The code is provided as-is and there is no guarantee that it is without bugs. If you do find a bug, please contact me.

The code is written as a MATLAB script instead of a function because I thought it would make learning easier because it can be stepped through, and all of the variables created by the code appear in the workspace so users can examine and manipulate them. If you don't want all of the variables in your workspace, it is straightforward to add a line at the top to turn it into a function. If you don't want all the intermediate results printed, simply comment out the lines you don't want.

There is essentially no error checking, so if you enter a netlist that isn't correct the program will fail without explaining why.

Notational conventions:

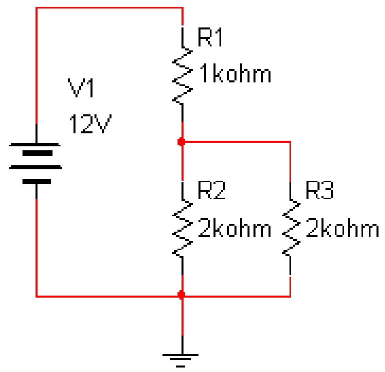
- Ground is labeled as node 0.
- The other nodes are labeled consecutively from 1 to *n*.
- We will refer to the voltage at node 1 as v_1, at node 2 as v_2 and so on.
- The naming of the independent voltage sources is quite loose, but the

names must start with the letter "V" and must be unique from any node names. For our purposes we will require that independent voltage sources have no underscore ("_") in their names. So the names Va, Vsource, V1, Vxyz123 are all legitimate names, but V_3, V_A, Vsource_1 are not.
- The current through a voltage source will be labeled with "I_" followed by the name of the voltage source. Therefore the current through Va is I_Va, the current through VSource is I_VSource, etc...
- The naming of the independent current sources is similar; the names must start with the letter "I" and must have no underscore ("_") in their names. So the names Ia, Isource, I1, Ixyz123 are all legitimate names, but I_3, I_A, Isource_1 are not.
- *Note: some of these rules are arbitrary (particularly the naming conventions) but make development of a MATLAB based algorithm easier.*

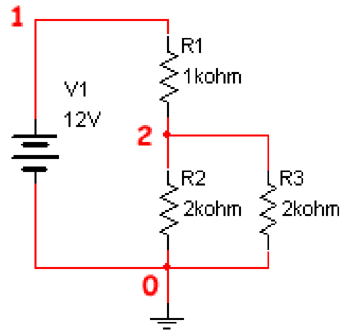
Defining circuits for SCAM: the netlist

The SCAM program cannot simply read a schematic diagram so we need to develop a method for representing a circuit textually. This can be done using a device called a netlist that defines the interconnection between circuit elements. If you have used SPICE (Simulation Program with Integrated Circuit Emphasis) this is a familiar concept. A good review is given by Jan Van Der Spiegel at the University of Pennsylvania. The process is easily demonstrated by example.

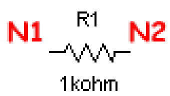
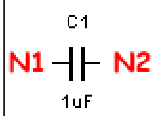
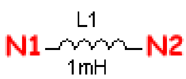
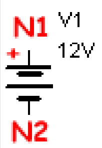
Let's use SCAM to analyze the circuit below:

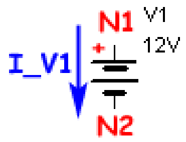
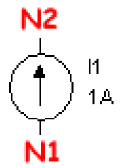
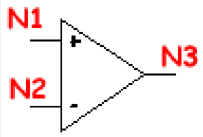


We start by defining the nodes. The only restriction here is that the nodes must be labeled such that ground is node 0, and the other nodes are named consecutively starting at 1. The choice of which number to assign to which node is entirely arbitrary.

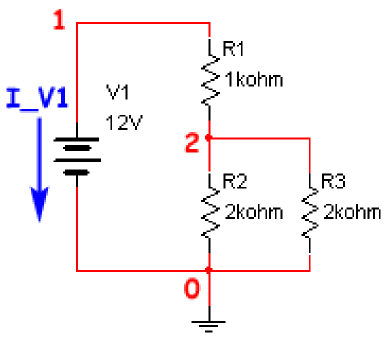


SCAM requires a text file with one line for each component in the circuit. This circuit has 4 components (3 resistors and 1voltage source), and will require 6 lines to define it. Each type of component has its own format for its corresponding lines in a file. These are shown below. The labels N1, N2, etc... correspond to the nodes in the circuit.

Component Type	Symbol	SCAM Description
Resistor		**R1 N1 N2 1000** • R1 is between nodes *N1* and *N2*, and has a value of 1000 Ohms • The value of the component must be written out (no abbreviations like kOhm) as a number. • The name of the component is R*x*, where *x* can be any combination of letters and numbers. R1, Rabc, Ra1 are all valid names.
Capacitor		**C1 N1 N2 1E-6** • Similar comments to the resistor
Inductor		**L1 N1 N2 1E-3** • Similar comments to the resistor
Voltage Source		**V1 N1 N2 12** • Similar comments to the resistor • Node *N1* is connected to the positive node, *N2* to the negative node. • The current through the

		source is one of the unknowns, it is defined as shown below: 
Current Source		**I1 N1 N2 1** • Similar comments to the resistor • Current flows out of node *N1* and into node *N2*.
Op Amp		**O1 N1 N2 N3** • Similar comments to the resistor but with three nodes as shown.

Define the current through the voltage source to give a final circuit diagram:

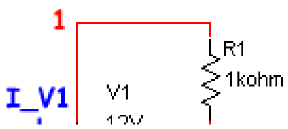


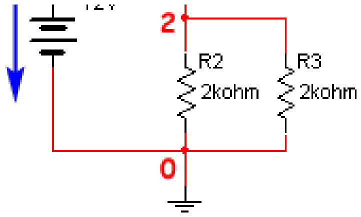
The netlist file is:

```
V1 1 0 12
R1 1 2 1000
R2 2 0 2000
R3 2 0 2000
```

Example 1: Voltages

Let's use the circuit we have been examining





We will create a text file containing the netlist:

```
V1 1 0 12
R1 1 2 1000
R2 2 0 2000
R3 2 0 2000
```

and save it in the directory seen by MATLAB. I edited such a file (using the MATLAB editor) and saved it in my SCAM directory as *example1.cir*. To run the program, assign the filename of the circuit to be analyzed to the variable *fname*, and then call the program. The output from the MATLAB window is shown below:

```
>> fname="example1.cir";
>> scam

Started -- please be patient.

Netlist:
V1 1 0 12
R1 1 2 1000
R2 2 0 2000
R3 2 0 2000

The A matrix:
[  1/R1,                -1/R1, 1]
[ -1/R1, 1/R1 + 1/R2 + 1/R3, 0]
[     1,                    0, 0]

The x matrix:
  v_1
  v_2
 I_V1

The z matrix:
  0
  0
 V1

The matrix equation:
            I_V1 + v_1/R1 - v_2/R1 == 0
 v_2*(1/R1 + 1/R2 + 1/R3) - v_1/R1 == 0
                              v_1 == V1

The solution:
                                       v_1 == V1
       v_2 == (R2*R3*V1)/(R1*R2 + R1*R3 + R2*R3)
 I_V1 == -(V1*(R2 + R3))/(R1*R2 + R1*R3 + R2*R3)

Elapsed time is 0.308426 seconds.
```

The netlist is displayed, followed by A, x and z matrices as well as the matrix equations written out. Finally the values of the unknow variables are displayed. Let's look at the value of v_2 (the voltage at node 2):

```
>> v_2
v_2 = (R2*R3*V1)/(R1*R2 + R1*R3 + R2*R3)
```

or I_V1 (the current through the voltage source)

```
>> I_V1
I_V1 = -(V1*(R2 + R3))/(R1*R2 + R1*R3 + R2*R3)
```

In addition to the unknowns, several other variables are created in the workspace (this is why the SCAM program is a *script* instead of a *function*). The important variables created, in addition to the unknowns, are a value corresponding to each of the elements. We can examine the value of any element, for example V1 or R2

```
>> V1
V1 =    12

>> R1
R1 =    1000
```

We can use these values to get numeric values for the unknowns:

```
>> eval(v_2)
ans =      6

>> eval(I_V1)
ans =    -0.0060
```

which shows that the voltage at node 2 is 6 volts, and the current through V1 is 6 mA (into the positive node).

Example 2: Currents

What happens if we are interested in the current through R2 instead of just node voltages, from node 2 to node 0. We know that the current through R2 is just the voltage drop across R2, divided by the value of the resistance. We can do the solution either symbolically or numerically.

```
>> v_2/R2
ans = (R2*R3*V1)/(2000*(R1*R2 + R1*R3 + R2*R3))

>> eval(ans)
ans =     0.0030
```

The current through R1, from node 1 to node 2, is just the voltage across R1 divided by its value.

```
>> (v_1-v_2)/R1
```

```
ans = V1/1000 - (R2*R3*V1)/(1000*(R1*R2 + R1*R3 + R2*R3))

>> eval(ans)
ans =     0.0060
```

Other quantities can be similarly determined. For example the ratio of v_2 to V1:

```
>> v_2/V1
ans = (R2*R3*V1)/(12*(R1*R2 + R1*R3 + R2*R3))

>> eval ans
ans = (R2*R3*V1)/(12*(R1*R2 + R1*R3 + R2*R3))
```

Example 3: Generating MNA Equations

The SCAM program also defines the **A**, **x** and **z** matrices from the MNA method.

```
>> A
A =
[  1/R1,              -1/R1, 1]
[ -1/R1, 1/R1 + 1/R2 + 1/R3, 0]
[     1,                  0, 0]

>> x
x =
  v_1
  v_2
 I_V1

>> z
z =
  0
  0
 V1
```

We can use these variables to recreate the circuit equations. To get the left side of the equations we just multiply A*X:

```
>> A*x
ans =
            I_V1 + v_1/R1 - v_2/R1
 v_2*(1/R1 + 1/R2 + 1/R3) - v_1/R1
                               v_1
```

or, in a slightly easier to read form:

```
>> pretty(A*x)
/             v_1    v_2       \
|      I_V1 + --- - ---        |
|             R1     R1        |
|                              |
|      /  1     1     1 \   v_1 |
| v_2 | -- + -- + -- | - --- |
|      \ R1    R2    R3 /   R1 |
|                              |
```

```
\             v_1             /
```

We can even get the Latex:

```
>> latex(A*x)
ans =
    '\left(\begin{array}{c} I_{\mathrm{V1}}+\frac{v_{1}}{I
```

The left side of the equation is given by **z**:

```
>> z
z =
  0
  0
 V1
```

Using the information above, we can get any of the MNA equations. To get the equations for node 2, simply take the 2nd row of the right and left sides of the equations:

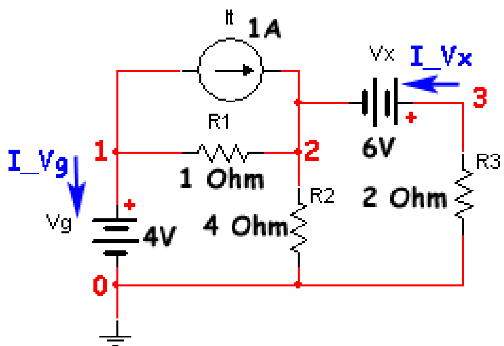
$$-\frac{v\_1}{R1}+\left(\frac{1}{R1}+\frac{1}{R2}+\frac{1}{R3}\right)v\_2=0$$

$$or$$

$$\left(\frac{v\_2-v\_1}{R1}\right)+\frac{v\_2}{R2}+\frac{v\_2}{R3}=0$$

Example 4: A More Complex Circuit

We can also apply the program to more complex circuits, such as the following (Example 3 from the MNA Examples page with values given to each component) (with nodes already labeled, and the currents through the voltage sources also labeled for clarity):



The netlist for this circuit is given by

```
Vg 1 0 4
Vx 3 2 6
R1 1 2 1
R2 2 0 4
R3 3 0 2
It 1 2 1
```

I entered this into a file and named it *example4.cir*. To analyze the circuit we proceed as before, by setting the *fname* variable, and starting the program:

```
>> fname='example4.cir';
>> scam

Started -- please be patient.

Netlist:
Vg 1 0 4
Vx 3 2 6
R1 1 2 1
R2 2 0 4
R3 3 0 2
It 1 2 1

The A matrix:
[  1/R1,       -1/R1,    0, 1,  0]
[ -1/R1, 1/R1 + 1/R2,    0, 0, -1]
[     0,           0, 1/R3, 0,  1]
[     1,           0,    0, 0,  0]
[     0,          -1,    1, 0,  0]

The x matrix:
  v_1
  v_2
  v_3
 I_Vg
 I_Vx

The z matrix:
 -It
  It
   0
  Vg
  Vx

The matrix equation:
           I_Vg + v_1/R1 - v_2/R1 == -It
 v_2*(1/R1 + 1/R2) - v_1/R1 - I_Vx == It
                     I_Vx + v_3/R3 == 0
                              v_1 == Vg
                        v_3 - v_2 == Vx

The solution:

            v_2 == (R2*R3*Vg - R1*R2*Vx + It*R1*R2*R3)/(I
        v_3 == (R3*(R2*Vg + R1*Vx + R2*Vx + It*R1*R2))/(I
 I_Vg == -(R2*Vg + R3*Vg + R2*Vx + It*R1*R2 + It*R1*R3)/(I
           I_Vx == -(R2*Vg + R1*Vx + R2*Vx + It*R1*R2)/(I

Elapsed time is 0.48674 seconds.
```

We can solve for the voltage at node 2 either symbolically or numerically:

```
>> v_2
v_2 = (R2*R3*Vg - R1*R2*Vx + It*R1*R2*R3)/(R1*R2 + R1*R3 +

>> eval(v_2)
ans =   1.1429
```

We can find the current through R1 (symbolically or numerically):

```
>> (v_1-v_2)/R1
ans =
Vg - (R2*R3*Vg - R1*R2*Vx + It*R1*R2*R3)/(R1*R2 + R1*R3 +
>> eval(ans)
ans =
    2.8571
```

We can find the MNA equation for node 2:

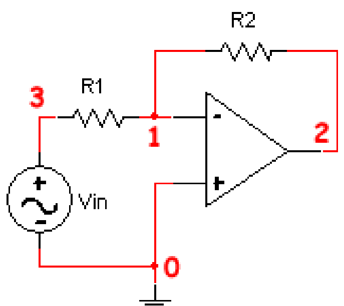
```
>> q=A*x==z
q =
            I_Vg + v_1/R1 - v_2/R1 == -It
 v_2*(1/R1 + 1/R2) - v_1/R1 - I_Vx == It
                   I_Vx + v_3/R3 == 0
                            v_1 == Vg
                      v_3 - v_2 == Vx

>> q(2)
ans = v_2*(1/R1 + 1/R2) - v_1/R1 - I_Vx == It

>> pretty(q(2))
    /  1     1 \    v_1
v_2 | -- + -- | - --- - I_Vx == It
    \ R1    R2 /    R1
```

or, in a cleaner format

$$v_2\left(\frac{1}{R_1}+\frac{1}{R_2}\right)-\frac{v_1}{R_1}-I_{\mathrm{Vx}}=\mathrm{It}$$

Example 5: Op Amps w/ resistors

The next example shows an op-amp with 2 resistors in the standard inverting configuration along with its netlist (*example5.cir*).

Circuit	Netlist
	Vin 3 0 Symbolic R1 1 3 Symbolic R2 2 1 Symbolic OAmp 0 1 2

Since we don't have values for the components in the circuit, they are declared to be "Symbolic". We can now solve this circuit to determine the gain between Vin and node 2.

```
>> fname="example5.cir";
>> scam
```

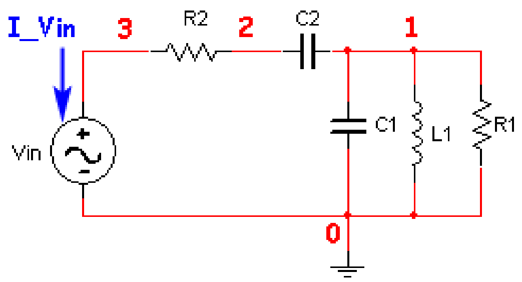
```
Started -- please be patient.

Netlist:
Vin 3 0 Symbolic
R1 1 3 Symbolic
R2 2 1 Symbolic
OAmp 0 1 2

The A matrix:
[ 1/R1 + 1/R2, -1/R2, -1/R1, 0, 0]
[       -1/R2,  1/R2,     0, 0, 1]
[       -1/R1,     0,  1/R1, 1, 0]
[           0,     0,     1, 0, 0]
[          -1,     0,     0, 0, 0]

The x matrix:
    v_1
    v_2
    v_3
  I_Vin
 I_OAmp

The z matrix:
   0
   0
   0
 Vin
   0

The matrix equation:
 v_1*(1/R1 + 1/R2) - v_2/R2 - v_3/R1 == 0
            I_OAmp - v_1/R2 + v_2/R2 == 0
             I_Vin - v_1/R1 + v_3/R1 == 0
                              v_3 == Vin
                               -v_1 == 0

The solution:
            v_1 == 0
 v_2 == -(R2*Vin)/R1
          v_3 == Vin
    I_Vin == -Vin/R1
    I_OAmp == Vin/R1

Elapsed time is 0.423071 seconds.

>> v_2/Vin
ans = -R2/R1
```

Some notes on this circuit:

- If a value is not given for an element, the value can be declared symbolic (as above). However this is not necessary, the user must simply ensure that the value is not a number -- so instead of "Symbolic", the value of R1 could have been "zyx" or even left blank.
- Because of the way we have used MNA to handle op-amps, the circuit above would give the exact same results even if the input terminals were switched. In practice this would not work. It is the responsibility of the user to make sure negative feedback exists.

Example 6: Generate a MATLAB Transfer Function Object.

Often we would like to take our results and use them in MATLAB for other calculations. this is especially true when working with transfer functions. The example below shows how this can be accomplished. This circuit is from the MNA with Capacitors and Inductors page with values given to each component, with nodes already labeled, and the currents through the voltage sources also labeled for clarity. The netlist is called *example6.cir*.

<table>
<tr><th>Circuit</th><th>Netlist</th></tr>
<tr><td>
</td><td>Vin 3 0 Symbolic
R2 3 2 1000
R1 1 0 1000
C1 1 0 1E-6
C2 2 1 10E-6
L1 1 0 0.001</td></tr>
</table>

Now let's solve and get the transfer function symbolically

```
>> fname="example6.cir";
>> scam

Started -- please be patient.

Netlist:
Vin 3 0 Symbolic
R2 3 2 1000
R1 1 0 1000
C1 1 0 1E-6
C2 2 1 10E-6
L1 1 0 0.001

The A matrix:
[ C1*s + C2*s + 1/R1 + 1/(L1*s),        -C2*s,      0, 0]
[                         -C2*s, C2*s + 1/R2, -1/R2, 0]
[                             0,        -1/R2,  1/R2, 1]
[                             0,            0,     1, 0]

The x matrix:
   v_1
   v_2
   v_3
 I_Vin

The z matrix:
   0
   0
   0
 Vin

The matrix equation:
 v_1*(C1*s + C2*s + 1/R1 + 1/(L1*s)) - C2*s*v_2 == 0
```

```
                 v_2*(C2*s + 1/R2) - v_3/R2 - C2*s*v_1 == 0
                              I_Vin - v_2/R2 + v_3/R2 == 0
                                                v_3 == Vin

The solution:
                                v_1 == (C2*L1*R1*Vin*s^2)/
  v_2 == (Vin*(R1 + L1*s + C1*L1*R1*s^2 + C2*L1*R1*s^2))/

 I_Vin == -(Vin*(C1*C2*L1*R1*s^3 + C2*L1*s^2 + C2*R1*s))/

Elapsed time is 0.380204 seconds.

>> H=v_2/Vin                %Find transfer function between
H =
(R1 + L1*s + C1*L1*R1*s^2 + C2*L1*R1*s^2)/(R1 + L1*s + C1*

>> H=collect(H)
H =
((C1*L1*R1 + C2*L1*R1)*s^2 + L1*s + R1)/(C1*C2*L1*R1*R2*s

>> pretty(H)             % Pretty print it.
                                            2
                      (C1 L1 R1 + C2 L1 R1) s  + L1 s + R1
----------------------------------------------------------
                 3                                   2
C1 C2 L1 R1 R2 s  + (C1 L1 R1 + C2 L1 R1 + C2 L1 R2) s  +
```

We can also get a numerical transfer function

```
>> Hnumbers = eval(H)
Hnumbers =
((6493253913945763*s^2)/590295810358705651712 + s/1000 + 1
```

While this answer is correct, it is not in a very convenient form, and we can't do any actual simulation with it. However we can easily convert the expression to a MATLAB transfer function object. First we separate out the numerator and denominator, and then convert them to MATLAB polynomials.

```
>> Hnumbers = eval(H)
Hnumbers =
((6493253913945763*s^2)/590295810358705651712 + s/1000 + 1

>> [n,d]=numden(Hnumbers)
n =
2536427310135063671875*s^2 + 230584300921369395200000*s +
d =
2305843009213693952*s^3 + 4842270319348757812500*s^2 + 230

>> mySys=tf(sym2poly(n),sym2poly(d))
mySys =
         2.536e21 s^2 + 2.306e23 s + 2.306e29
  ---------------------------------------------------
  2.306e18 s^3 + 4.842e21 s^2 + 2.306e27 s + 2.306e29

Continuous-time transfer function.
```

The transfer function is still in an odd form (we'd like the highest powers of 's' in the denominator to have a coefficient of 1). Fortunately, MATLAB's "`minreal()`"

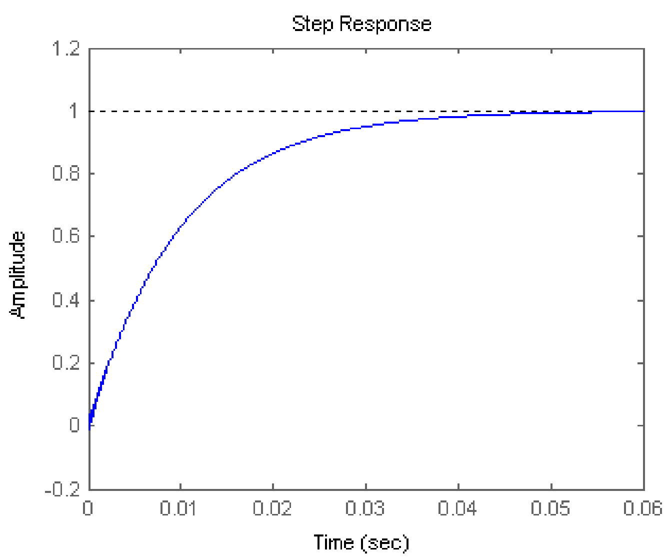
function does this.

```
>> mySys=minreal(mySys)
mySys =
    1100 s^2 + 100000 s + 1e11
  ------------------------------
  s^3 + 2100 s^2 + 1e09 s + 1e11

Continuous-time transfer function.
```

We are now free to perform any of the MATLAB function relating to polynomials. Shown below are a step response and a Bode plot.

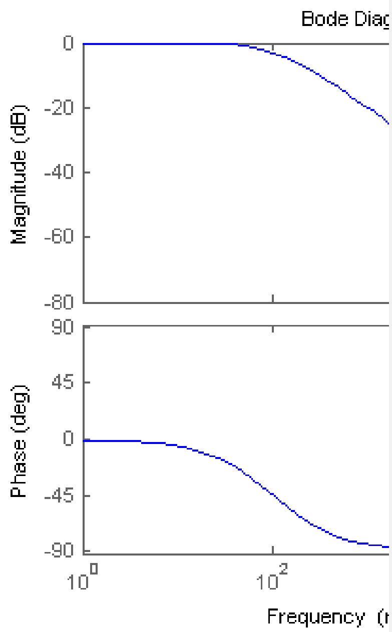
Step Response

```
>> step(mySys)
```



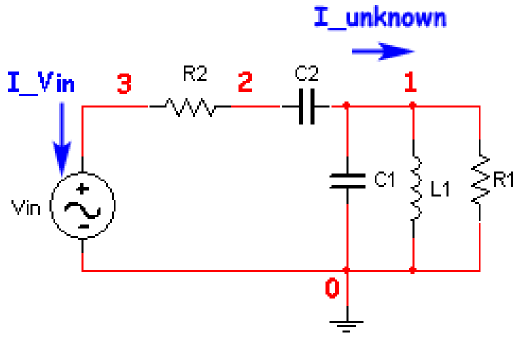
Bode Plot

```
>>bode(mySys)
```

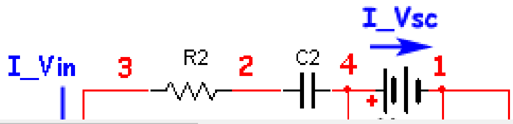


Example 7: Finding the current in a wire

Now consider the case of finding the current through a wire. In particular, consider the previous circuit:



We would like to find the current shown. How do we do this. One way would be to find each current into and out of node 1 and solve for the unknown current. Another way is to introduce a voltage source of zero volts (i.e., a short circuit), and its concomitant node. The netlist is *example7.cir*.

Circuit	Netlist
	Vin 3 0 Symbolic R2 3 2 1000 R1 1 0 1000

Advanced SCAM with dependent sources

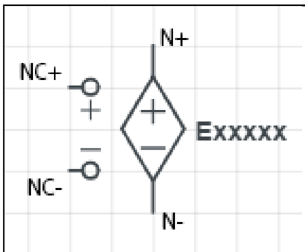
Contents

This document gives a brief overview of how dependent sources are accommodated in modified nodal analysis. The description of each type of source is brief, but is followed by an example. Between the description and the example you should be able to figure out how to add dependent sources to your model. You may want to review the MNA algorithm if necessary (especially the role of the various matrices and submatrices), as well as how to use the SCAM program. Note: the example netlists (as well as the MATLAB code) are on GitHub

Voltage Controlled Voltage Source (VCVS) - E

Voltage controlled voltage sources (VCVS) are defined in a netlist with an element name (that begins with the letter "E"), followed by the two output nodes (N+ and N-), the two controlling nodes (NC+ and NC-) and the gain Value. The output voltage is set to the gain Value multiplied by the difference between the controlling nodes.

$$(v_{N+} - v_{N-}) = Value \cdot (v_{NC+} - v_{NC-})$$



`Exxxxx N+ N- NC+ NC- Value`

Since the VCVS adds a voltage source to the circuit we also need to add a new unknown, which is the current throught the voltage source (so the **B**, **C**, and **D** submatrices all become larger — review the MNA algorithm if necessary). Recall that currents out of the node are taken to be positive.

Changes to MNA Matrices

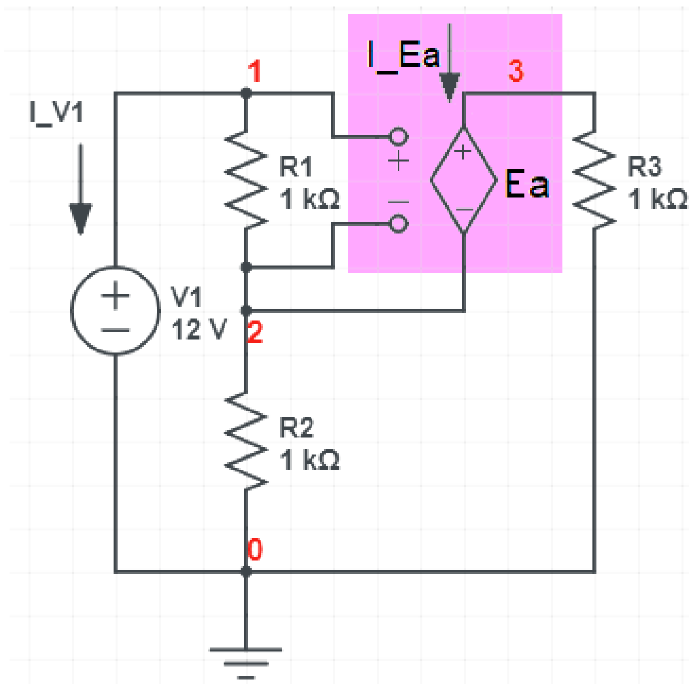
- **j** matrix: Since the output is a voltage source we add the unknown current through the source (I_Exxxxx) to the matrix **j**.
- **e** matrix: Add a row with a zero in it to account for the new dependent voltage source.
- **B** matrix: We add a column to the B matrix for the new voltage source and add +1 and -1 in the row numbered "N+" and "N-" to account for the current added to (or subtracted from) that node.
- **C** matrix: We add a row for the new voltage source. We can rewrite the equation above as

$$(v_{N+} - v_{N-}) + Value \cdot (v_{NC-} - v_{NC+}) = 0$$

so we add +1 in the column corresponding to "N+" and "-1" in the column corresponding to "N-". We also add "+Value" (the gain term) in the column corresponding to "NC-" and "-Value" for "NC+".

Example:

The VCVS is demonstrated in the circuit shown



We can analyze with scam and you can see how the VCVS effects the matrices (the netlist is shown near the top of the output).

```
>> scam

Netlist:
V1 1 0 12
R1 1 2 1000
R2 2 0 1000
R3 3 0 1000
Ea 3 2 1 2 g

The A matrix:
[  1/R1,        -1/R1,     0, 1,  0]
[ -1/R1, 1/R1 + 1/R2,     0, 0, -1]
```

```
[      0,            0, 1/R3, 0,  1]
[      1,            0,    0, 0,  0]
[    -Ea,       Ea - 1,    1, 0,  0]

The x matrix:
  v_1
  v_2
  v_3
 I_V1
 I_Ea

The z matrix:
  0
  0
  0
 V1
  0

The matrix equation:
            I_V1 + v_1/R1 - v_2/R1 == 0
 v_2*(1/R1 + 1/R2) - v_1/R1 - I_Ea == 0
                     I_Ea + v_3/R3 == 0
                              v_1 == V1
      v_3 - Ea*v_1 + v_2*(Ea - 1) == 0

The solution:
                                                    v_1
 v_2 == (R2*V1*(R3 - Ea*R1))/(R1*R2 + R1*R3 + R2*R3 - Ea*I
 v_3 == (R3*V1*(R2 + Ea*R1))/(R1*R2 + R1*R3 + R2*R3 - Ea*I
     I_V1 == -(V1*(R2 + R3))/(R1*R2 + R1*R3 + R2*R3 - Ea*I
  I_Ea == -(V1*(R2 + Ea*R1))/(R1*R2 + R1*R3 + R2*R3 - Ea*I
```

If you look carefully you will see that a variable has been added to the **x** matrix for the current through the VCVS (labelled "I_Ea") and that the last row of the **A** matrix defines the relationship between input (controlling) and output voltages for the VCVS, effectively:

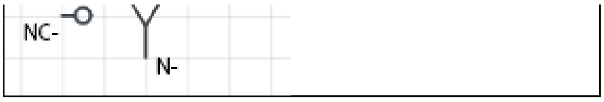
$$(v\_3 - v\_2) = Ea \cdot (v\_1 - v\_2)$$

Voltage Controlled Current Source (VCCS) - G

Voltage controlled current sources (VCCS) are defined in a netlist with an element name (that begins with the letter "G"), followed by the two output nodes (N+ and N-), the two controlling nodes (NC+ and NC-) and the gain Value. The output current is set to the gain Value multiplied by the difference between the controlling nodes. The circuit equations for nodes N+ and N- (i.e., summing the current at each node to zero) must be changed to account for the new current:

$$\begin{aligned} \text{node N+ :} \qquad & Value \cdot (v_{NC+} - v_{NC-}) + \sum(\text{other currents into N+}) = 0 \\ \text{node N- :} \qquad & -Value \cdot (v_{NC+} - v_{NC-}) + \sum(\text{other currents into N-}) = 0 \end{aligned}$$

Gxxxxx N+ N- NC+ NC- Value



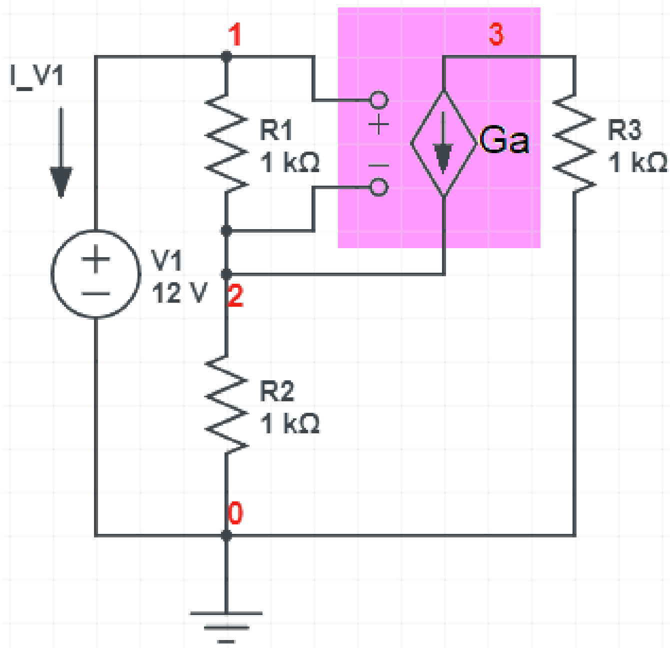
Recall that currents out of the node are taken to be positive.

Changes to MNA Matrices

The only matrix that changes is the **G** matrix. If you examine the defining equations above you see that in the row of **G** associated with "N+" we add "+Value" to the column associated with "NC+" and "-Value" to the node associated with "NC-". In the row of **G** associated with "N-" we add "-Value" to the column associated with "NC+" and "+Value" to the node associated with "NC-".

Example:

The VCCS is demonstrated in the circuit shown



We can analyze with scam and you can see how the VCCS effects the matrices (the netlist is shown near the top of the output).

```
>> scam

Netlist:
V1 1 0 12
R1 1 2 1000
R2 2 0 1000
R3 3 0 1000
Ga 3 2 1 2 g

The A matrix:
[          1/R1,            -1/R1,    0, 1]
[ - Ga - 1/R1, Ga + 1/R1 + 1/R2,    0, 0]
[            Ga,              -Ga, 1/R3, 0]
[             1,                0,    0, 0]

The x matrix:
  v_1
  v_2
  v_3
 I_V1
```

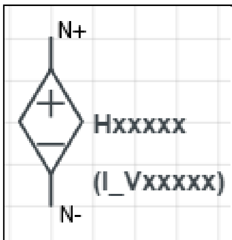
```
 I_V1

The z matrix:
  0
  0
  0
 V1

The matrix equation:
                   I_V1 + v_1/R1 - v_2/R1 == 0
 v_2*(Ga + 1/R1 + 1/R2) - v_1*(Ga + 1/R1) == 0
                 Ga*v_1 - Ga*v_2 + v_3/R3 == 0
                                    v_1 == V1

The solution:
                                      v_1 == V1
 v_2 == (R2*V1*(Ga*R1 + 1))/(R1 + R2 + Ga*R1*R2)
      v_3 == -(Ga*R1*R3*V1)/(R1 + R2 + Ga*R1*R2)
               I_V1 == -V1/(R1 + R2 + Ga*R1*R2)
```

Since no new voltage sources are added to the circuit, the matrices don't change size, however the **G** submatrix must be altered to account for the current from the VCCS.

Current Controlled Voltage Source (CCVS) - H

For the current controlled voltage source (CCCS) the controlling quantity is current. You will recall that the only currents that are calculated by the MNA algorithm are currents through voltage sources. Therefore, the controlling current must be the current through one of the voltage sources. It is this voltage source that is used in the netlist as the controlling element — but keep in mind it is the *current* through voltage source that is important. *(Note: if you reqquire a controlling current that is not initailly the current through a voltage source, you need to add a zero voltage source to the circuit (see example) so that current will be calculated.)*. The CCVS is defined in a netlist with an element name (that begins with the letter "H"), followed by the two output nodes (N+ and N-), the name of the voltage source whose current is the controlling current, and the gain Value. For this element, the output voltage is set to the gain Value multiplied by the controlling current.

$$(v_{N+} - v_{N-}) = Value \cdot I\_Vxxxxx$$




```
Hxxxxx N+ N- Vxxxxx Value
```

Since the CCVS adds a voltage source to the circuit we also need to add a new unknown, which is the current throught the voltage source (so the **B**, **C**, and **D** submatrices all become larger — review the MNA algorithm if necessary). Recall that currents out of the node are taken to be positive.

Changes to MNA Matrices

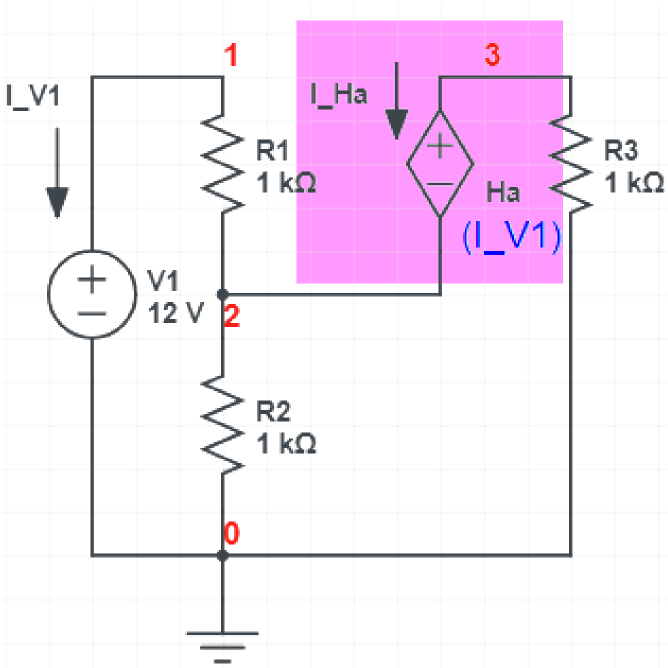
- I matrix: Since the output is a voltage source we add the unknown current

- **j** matrix: Since the output is a voltage source we add the unknown current through the source (I_Exxxxx) to the matrix **j**.
- **e** matrix: Add a row with a zero in it to account for the new dependent voltage source.
- **B** matrix: We add a column to the B matrix for the new voltage source and add +1 and -1 in the row numbered "N+" and "N-" to account for the current added to (or subtracted from) that node.
- **C** matrix: We add a row for the new voltage source and we add +1 in the column corresponding to "N+" and "-1" in the column corresponding to "N-".
- **D** matrix: In the row corresponding to the new voltage source, and the column corresponding to the controlling source we subtract "Value". This has the effect of making that row correspond to the equation:

$$(v_{N+} - v_{N-}) - Value \cdot I\_Vxxxxx = 0$$

Example:

The use of a CCVS is demonstrated in the circuit shown whose netlist is given by:



We can analyze with scam and you can see how the CCVS effects the matrices (the netlist is shown near the top of the output).

```
>> scam


Netlist:
V1 1 0 12
R1 1 2 1000
R2 2 0 1000
R3 3 0 1000
Ha 3 2 V1 r

The A matrix:
[  1/R1,        -1/R1,     0,    1,   0]
[ -1/R1,  1/R1 + 1/R2,     0,    0,  -1]
```

```
[ -1/R1, 1/R1 + 1/R2,    0,    0, -1]
[     0,           0, 1/R3,    0,  1]
[     1,           0,    0,    0,  0]
[     0,          -1,    1, -Ha,  0]

The x matrix:
  v_1
  v_2
  v_3
 I_V1
 I_Ha

The z matrix:
  0
  0
  0
 V1
  0

The matrix equation:
            I_V1 + v_1/R1 - v_2/R1 == 0
 v_2*(1/R1 + 1/R2) - v_1/R1 - I_Ha == 0
                     I_Ha + v_3/R3 == 0
                               v_1 == V1
               v_3 - v_2 - Ha*I_V1 == 0

The solution:
                                                                     v_1 == V1

  v_2 == (R2*V1*(Ha + R3))/(R1*R2 + R1*R3 + R2*R3 + Ha*R2)
 v_3 == -(R3*V1*(Ha - R2))/(R1*R2 + R1*R3 + R2*R3 + Ha*R2)
   I_V1 == -(V1*(R2 + R3))/(R1*R2 + R1*R3 + R2*R3 + Ha*R2)
    I_Ha == (V1*(Ha - R2))/(R1*R2 + R1*R3 + R2*R3 + Ha*R2)
```

As you can see the **B**, **C**, and **D** submatrices have been enlarged to account for the new unknown variable (I_Ha, the current through the CCVS output), and the **D** matrix has been altered such that the last equation is effectively:

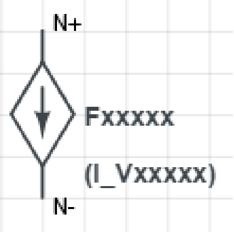
$$(v\_3 - v\_2) = Ha \cdot I\_V1$$

as desired.

Current Controlled Current Source (CCCS) - F

For the current controlled current source (CCCS) the controlling quantity is current. You will recall that the only currents that are calculated by the MNA algorithm are currents through voltage sources. Therefore, the controlling current must be the current through one of the voltage sources. It is this voltage source that is used in the netlist as the controlling element — but keep in mind it is the *current* through voltage source that is important. *(Note: if you reqquire a controlling current that is not initailly the current through a voltage source, you need to add a zero voltage source to the circuit (see example) so that current will be calculated.)*. The CCCS is defined in a netlist with an element name (that begins with the letter "F"), followed by the two output nodes (N+ and N-), the name of the voltage source whose current is the controlling current, and the gain Value. For this element, the output current is set to the gain Value multiplied by the controlling current. The circuit equations for nodes N+ and N- (i.e., summing the current at each node to zero) must be changed to account for the new current (in this equation "I_Vxxxxx" is the current through the voltage source "Vxxxxx" (defined elsewhere in the netlist):

$$\begin{aligned} &\text{node N+ :} && Value \cdot I\_Vxxxxx + \sum(\text{other currents into N+}) = 0 \\ &\text{node N- :} && -Value \cdot I\_Vxxxxx + \sum(\text{other currents into N-}) = 0 \end{aligned}$$




```
Fxxxxx N+ N- Vxxxxx Value
```

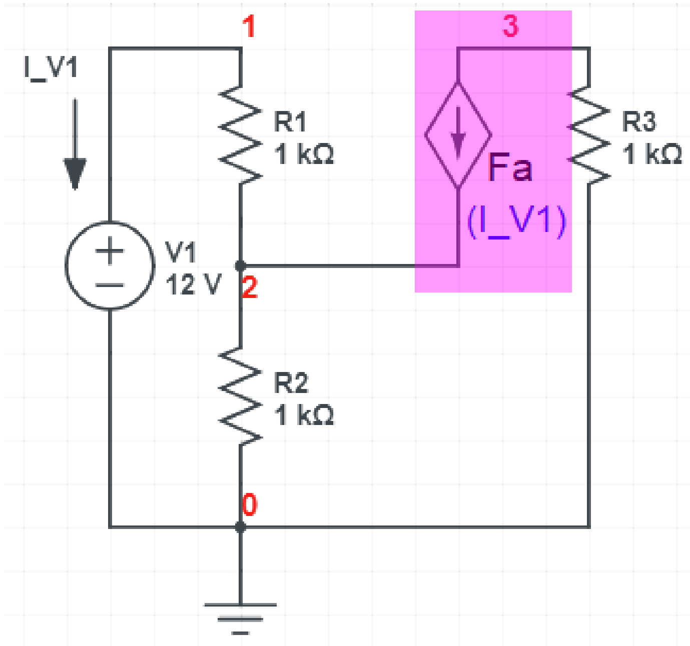
Recall that currents out of the node are taken to be positive.

Changes to MNA Matrices

Only the **B** matrix changes. For this element we need to add a current equal to "Value·I_Vxxxxx" to node "N+" and subtract it from node "N-". To do this we find the column of **B** that correspondes to the voltage source "Vxxxxx". To the row "N+" of this column we add "Value" (so that it multiplies "I_Vxxxxx") which effectively adds "Value·I_Vxxxxx" to that node . Likewise, we subtract "Value" from the row "N-" of this same column of **B**.

Example:

The use of a CCCS is demonstrated in the circuit shown whose netlist is given by (in this case the controlling current is the current through "V1"):



We can analyze with scam and you can see how the CCCS effects the matrices (the netlist is shown near the top of the output).

```
>> scam

Netlist:
V1 1 0 12
R1 1 2 1000
```

```
R2 2 0 1000
R3 3 0 1000
Fa 3 2 V1 f

The A matrix:
[  1/R1,        -1/R1,     0,    1]
[ -1/R1, 1/R1 + 1/R2,     0,  -Fa]
[     0,            0, 1/R3,   Fa]
[     1,            0,     0,    0]

The x matrix:
  v_1
  v_2
  v_3
 I_V1

The z matrix:
  0
  0
  0
 V1
```

Referei