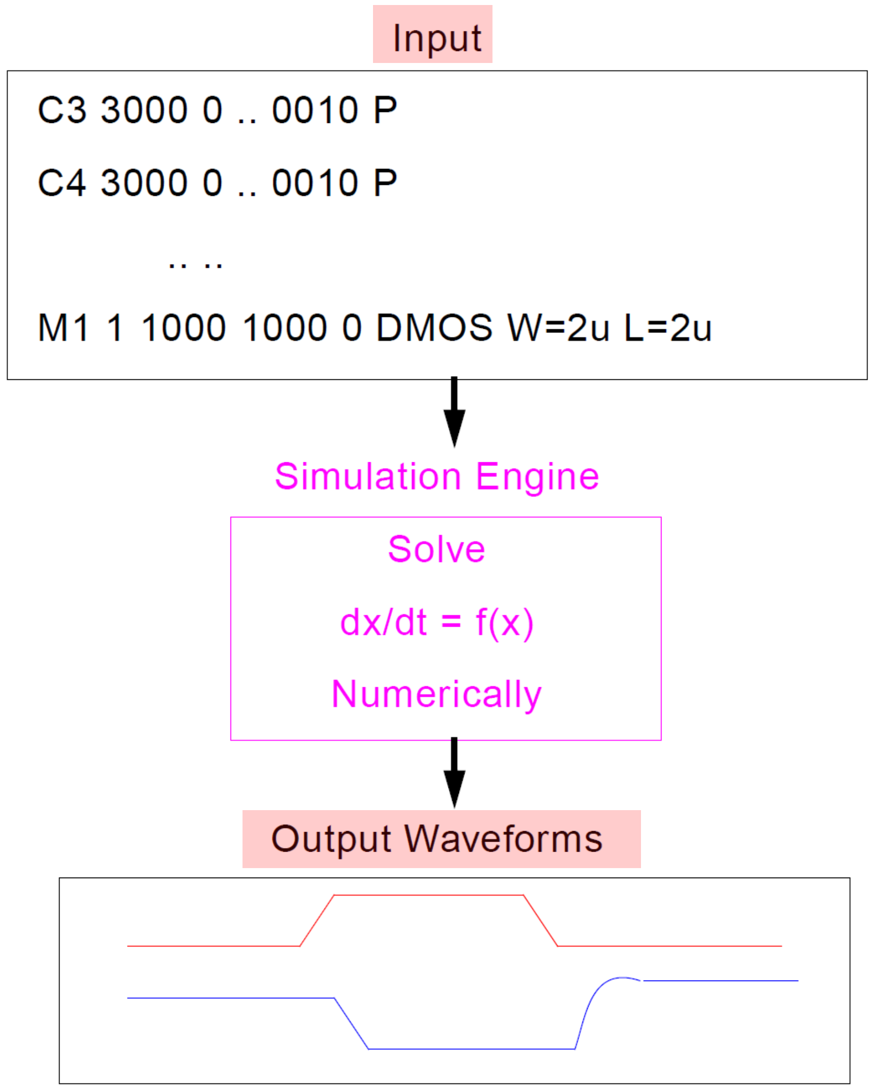
CIRCUIT SIMULATION-1
CIRCUIT NETWORK ANALYSIS

Circuit Simulation

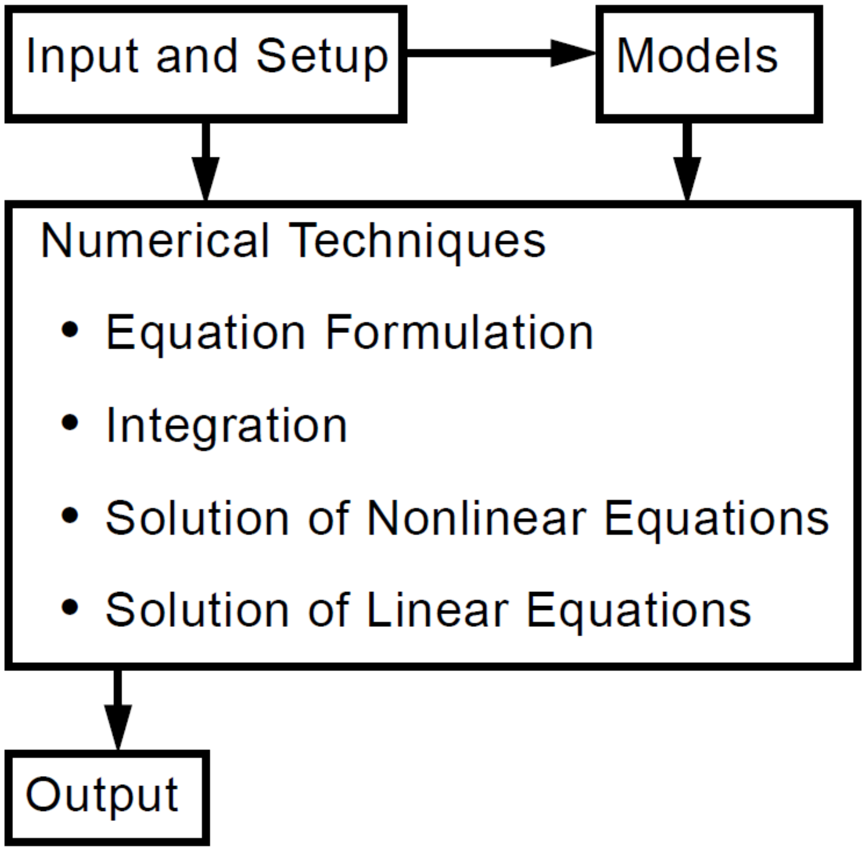
Input: **Circuit description and input stimulus**

Output: **Approximate voltage or current waveforms.**

T. L. Pillage, R. A. Rohrer, and C. Visweswariah, *Electronic Circuit & System Simulation Methods*, McGraw-Hill, Inc. 1995.



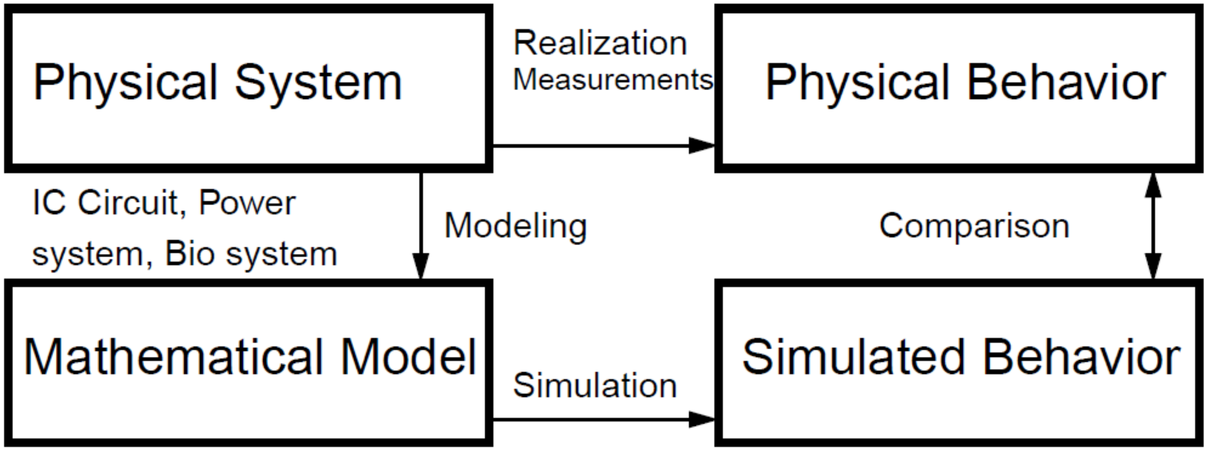
Simulation Program Structure



Types of Analysis

- **DC Analysis**
- **DC Transfer Curves**
- **Transient Analysis**
- **AC Analysis Noise, Distortion, Sensitivity**

Modeling



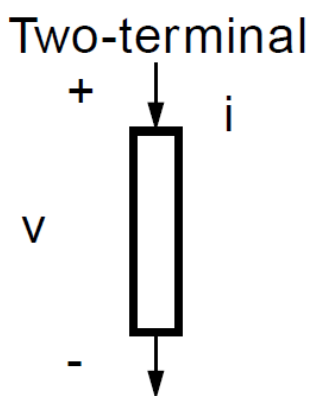
Formulation of Circuit Equations

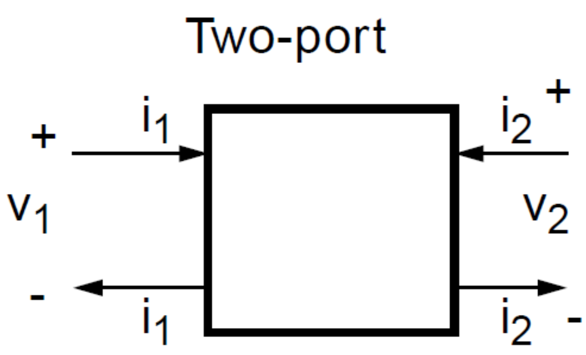
- **KCL (Kirchoff's Current Law): branch currents sum to zero at every node.**
- **KVL (Kirchoff's Voltage Law): branch voltage sum to zero around every loop.**
- **BCR (Branch Constitutive Relationships): *i.e.*, device models, or equations for all the circuit components.**

BCR Branch Equations

Mathematical models of circuit components are expressed:

- **in terms of ideal elements --Inductors, Capacitors, Resistors, Current Sources, Voltage Sources, Two-ports, ...**
- **in terms of physical quantities - current, voltage, charge, flux**

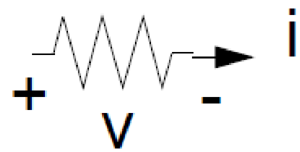




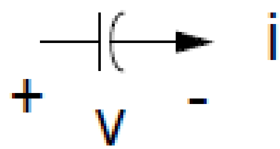
where i, v are branch currents and voltages respectively.

Ideal Elements: Linear Two-Terminal

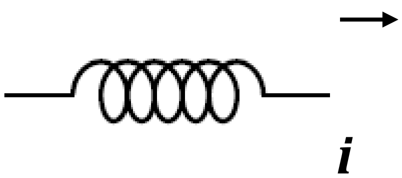
- **Resistor**: $i = (1/R)v$; $v = R\,i$



- **Capacitor**: $q = C\,v$; $i = dq/dt = C\,dv/dt$

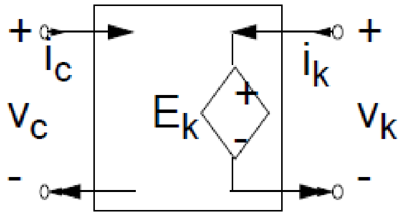


- **Inductor**: $v = L\,di/dt$

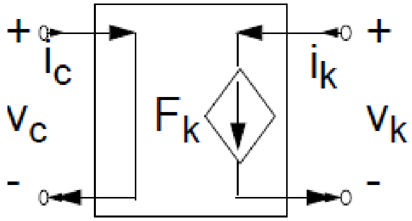


Controlled Sources

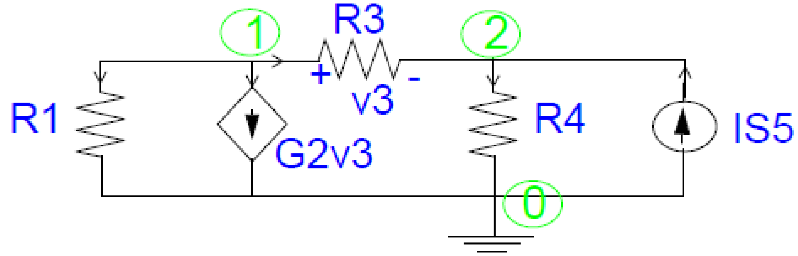
- **VCVS (Voltage Controlled Voltage Source):** $v_k = E_k v_{c;} \quad i_c = 0.$



- **CCCS (Current Controlled Current Source):** $i_k = F_k i_{c;} \quad v_c = 0.$



Matrix Forms of KVL and KCL



- **KCL:** $\boldsymbol{A}\,\boldsymbol{i}=\mathbf{0}$

$$\begin{bmatrix} 1 & 1 & 1 & 0 & 0 \\ 0 & 0 & -1 & 1 & -1 \end{bmatrix} \begin{bmatrix} i_1 \\ i_2 \\ i_3 \\ i_4 \\ i_5 \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \end{bmatrix}$$

- **KVL:** $\boldsymbol{v} - \boldsymbol{A}^T\boldsymbol{e} = \mathbf{0}$

$$\begin{bmatrix} v_1 \\ v_2 \\ v_3 \\ v_4 \\ v_5 \end{bmatrix} - \begin{bmatrix} 1 & 0 \\ 1 & 0 \\ 1 & -1 \\ 0 & 1 \\ 0 & -1 \end{bmatrix} \begin{bmatrix} e_1 \\ e_2 \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \\ 0 \\ 0 \\ 0 \end{bmatrix}$$

Equation Assembly for Linear Circuits

(1) Sparse Tableau Analysis (STA)

(2) Modified Nodal Analysis (MNA)

(3) Tree/link Analysis

(1) STA: Sparse Tableau Analysis [Brayton, Gustavson, Hachtel (1969-71)]

- **Write KCL:** $A\,i = 0,$ n equations for each node
- **Write KVL:** $v - A^T e = 0,$ b equations for each branch
- **Write Branch Equations:**

  $K_i\, i$ for current controlled;

  $K_v\, v$ for voltage controlled.

  $K_i\, i + K_v\, v = S;$ b equations

$$\begin{bmatrix} A & 0 & 0 \\ 0 & I & -A^T \\ K_i & K_v & 0 \end{bmatrix} \begin{bmatrix} i \\ v \\ e \end{bmatrix} = \left. \begin{bmatrix} 0 \\ 0 \\ S \end{bmatrix} \right\} \text{n + 2b equations}$$

sparse tableau

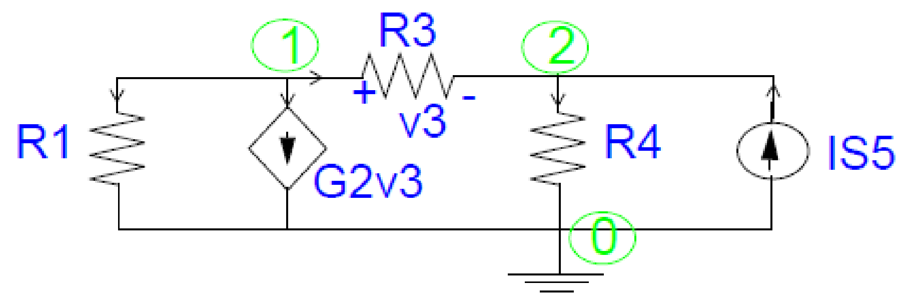
n + 2b unknowns

n = # nodes
b = # branches

Equation Assembly for Linear Circuits

(2) Nodal Analysis **[McCalla, Nagel, Rohrer, Ruehli, Ho]**

Step 1: **Write KCL**



$$\boldsymbol{i1 + i2 + i3 = 0}$$ **on node ①**

$$\boldsymbol{-i3 + i4 - i5 = 0}$$ **on node ②**

(I)

Step 2: **Use branch equations to eliminate branch currents from (I)**

$$\frac{1}{R1}v1 + G2v3 + \frac{1}{R3}v3 = 0$$ **node ①**

$$-\frac{1}{R3}v3 + \frac{1}{R4}v4 = IS5$$ **node ②** **(II)**

Step 3: **Use KVL to eliminate branch voltages from (II)**

$$\frac{1}{R1}e1 + G2(e1 - e2) + \frac{1}{R3}(e1 - e2) = 0$$ **node ①**

$$-\frac{1}{R3}(e1 - e2) + \frac{1}{R4}e2 = IS5$$ **node ②** **(III)**

Nodel Analysis in Matrix Form: Y_n e = IS

$$\begin{bmatrix} \frac{1}{R1} + G2 + \frac{1}{R3} & -G2 - \frac{1}{R3} \\ -\frac{1}{R3} & \frac{1}{R3} + \frac{1}{R4} \end{bmatrix} \begin{bmatrix} e1 \\ e2 \end{bmatrix} = \begin{bmatrix} 0 \\ IS5 \end{bmatrix}$$

Y_n and IS can be directly assembled from the input data.

Note that the system matrix can be written as the sum of 4 matrices:

$$\begin{bmatrix} \frac{1}{R1} + G2 + \frac{1}{R3} & -G2 - \frac{1}{R3} \\ -\frac{1}{R3} & \frac{1}{R3} + \frac{1}{R4} \end{bmatrix} = \begin{bmatrix} \frac{1}{R1} & 0 \\ 0 & 0 \end{bmatrix} + \begin{bmatrix} G2 & -G2 \\ 0 & 0 \end{bmatrix} + \begin{bmatrix} \frac{1}{R3} & -\frac{1}{R3} \\ -\frac{1}{R3} & \frac{1}{R3} \end{bmatrix} + \begin{bmatrix} 0 & 0 \\ 0 & \frac{1}{R4} \end{bmatrix}$$

Each matrix relates to a specific element.

The contribution of an element is called *element stamp*.

Nodal Analysis Elements (1)

Resistor: **R_k N+ N- $R_k$value** **(SPICE Input Format)**

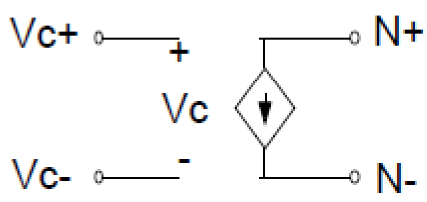
Resistor "Stamp":

$$\begin{array}{cc} & \begin{array}{ccc} N+ & \ldots & N- \end{array} \\ \begin{array}{c} N+ \\ \vdots \\ N- \end{array} & \left[\begin{array}{ccc} 1/R_k & & -1/R_k \\ \vdots & & \vdots \\ -1/R_k & & 1/R_k \end{array}\right] \end{array}$$

Nodal Analysis Elements (2)

VCCS: **G_k N+ N- NC+ NC- $G_k$value**

VCCS "Stamp":

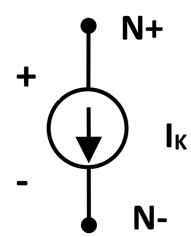


$$\begin{array}{c} \\ N+ \\ \vdots \\ N- \\ \\ \end{array} \begin{array}{c} \begin{array}{cc} V_{c+} & V_{c-} \end{array} \\ \left[\begin{array}{cc} G_k & -G_k \\ \vdots & \vdots \\ -G_k & G_k \\ & \end{array}\right] \end{array}$$

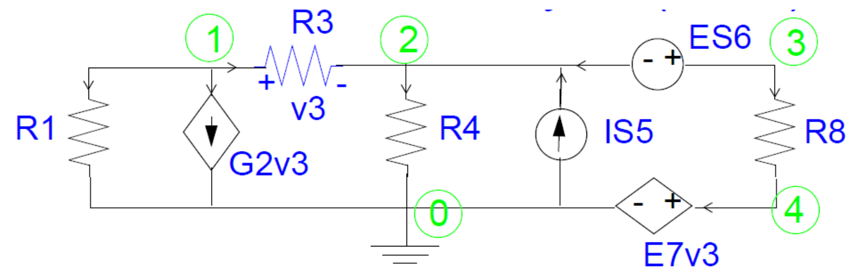
Independent Current Source: **I_k N+ N- $I_k$value**

Stamp:

$$\begin{array}{c} N+ \\ \vdots \\ N- \end{array} \left[\begin{array}{ccc} & & \\ & \vdots & \\ & & \end{array}\right] = \left[\begin{array}{c} -I_k \\ +I_k \end{array}\right]$$



MNA: Modified Nodal Analysis (1)



Step 1: Write KCL

$$i1 + i2 + i3 = 0$$ on node ①

$$-i3 + i4 - i5 - i6 = 0$$ on node ②

$$i6 + i8 = 0$$ on node ③

$$i7 - i8 = 0$$ on node ④

Step 2: Use branch equations to eliminate as many branch currents as possible

$$\frac{1}{R1}v1 + G2v3 + \frac{1}{R3}v3 = 0$$ node ①

$$-\frac{1}{R3}v3 + \frac{1}{R4}v4 - i6 = IS5$$ node ②

$$i6 + \frac{1}{R8}v8 = 0$$ node ③

$$\mathrm{i}7 - \frac{1}{R8}v8 = 0$$ node ④

MNA: Modified Nodal Analysis (2)

Step 3: **Write down unused branch equations**

$$\boldsymbol{v6 = ES6} \qquad \textbf{(b6)}$$

$$\boldsymbol{v7 - E7v3 = 0} \qquad \textbf{(b7)}$$

Step 4: **Use KVL to eliminate branch voltages from previous equations**

$$\frac{1}{R1}e1 + G2(e1 - e2) + \frac{1}{R3}(e1 - e2) = 0 \qquad \textbf{node ①}$$

$$-\frac{1}{R3}(e1 - e2) + \frac{1}{R4}e2 - i6 = IS5 \qquad \textbf{node ②}$$

$$i6 + \frac{1}{R8}(e3 - e4) = 0 \qquad \textbf{node ③}$$

$$i7 - \frac{1}{R8}(e3 - e4) = 0 \qquad \textbf{node ④}$$

$$(e3 - e2) = ES6 \qquad \textbf{(b6)}$$

$$e4 - E7(e1 - e2) = 0 \qquad \textbf{(b7)}$$

MNA in Matrix Form

$$\begin{bmatrix} \frac{1}{R1}+G2+\frac{1}{R3} & -G2-\frac{1}{R3} & 0 & 0 & 0 & 0 \\ -\frac{1}{R3} & \frac{1}{R3}+\frac{1}{R4} & 0 & 0 & -1 & 0 \\ 0 & 0 & \frac{1}{R8} & -\frac{1}{R8} & 1 & 0 \\ 0 & 0 & -\frac{1}{R8} & \frac{1}{R8} & 0 & 1 \\ 0 & -1 & 1 & 0 & 0 & 0 \\ E7 & -E7 & 0 & -1 & 0 & 0 \end{bmatrix} \begin{bmatrix} e1 \\ e2 \\ e3 \\ e4 \\ i6 \\ i7 \end{bmatrix} = \begin{bmatrix} 0 \\ IS5 \\ 0 \\ 0 \\ ES6 \\ 0 \end{bmatrix}$$

$$\left[\begin{array}{c|c} Y_n & B \\ \hline C & 0 \end{array}\right] \left[\begin{array}{c} e \\ \hline i \end{array}\right] = MS$$

node voltages

solve for some branch currents

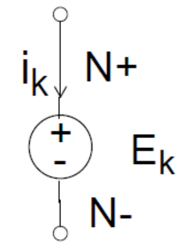
- **MNA matrix and MS can be directly assembled from the input by inspection.**

Nodal Analysis Elements (3)

Floating Voltage Source: ESK N+ N- E_k

Stamp:

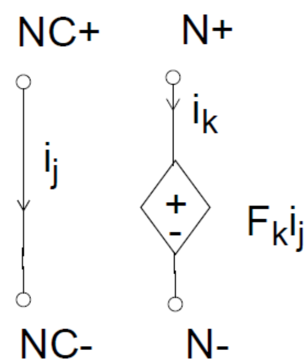
	N+	N-	i_k	RHS
N+	0	0	1	0
N-	0	0	-1	0
branch k	1	-1	0	E_k



CCVS (Current controlled voltage source): FSK N+ N- NC+ NC- F_k

Stamp:

	N+	N-	NC+	NC-	i_k	i_j
N+					1	
N-					-1	
NC+						1
NC-						-1
branch k	1	-1				$-F_k$
branch j			1	-1		



General Rules for MNA

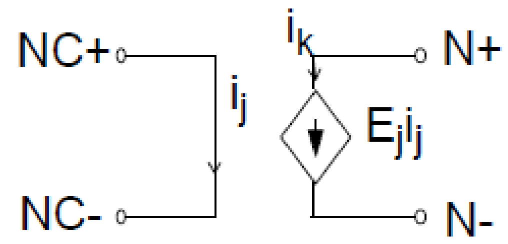
1. **A branch current is always introduced as an additional variable for a voltage source or an inductor.**
2. **For current sources, resistors, conductances, and capacitors, the branch current is introduced only if**
   - **any circuit element depends on that branch current**
   - **the branch current is requested as an output**

Nodal Analysis Elements (4)

CCCS: Current controlled current source

Stamp:

	N+	N-	NC+	NC-	i_j
N+					E_j
N-					$-E_j$
NC+					1
NC-					-1
branch j			1	-1	



Advantages of MNA

1. **MNA can be applied to any circuit.**
2. **MNA equations can be assembled “directly” (e.g., from SPICE input deck) from input data.**
3. **MNA matrix is close to Y_n.**

Circuit Simulation-2
Simulator Design

LU Decomposition (1)

Given Ax = b, decompose it into: A = LU, LUx = b

$$\begin{bmatrix} a_{11} & a_{12} & \dots & a_{1n} \\ a_{21} & a_{22} & \dots & a_{2n} \\ \vdots & & \ddots & \vdots \\ a_{n1} & a_{n2} & \dots & a_{nn} \end{bmatrix} = \begin{bmatrix} l_{11} & l_{11}u_{12} & l_{11}u_{13} & \dots \\ l_{21} & l_{21}u_{12} + l_{22} & l_{21}u_{13} + l_{22}u_{23} & \dots \\ \vdots & & \ddots & \vdots \\ l_{n1} & l_{n1}u_{12} + l_{n2} & l_{n1}u_{13} + l_{n2}u_{23} + l_{n3} & \dots \end{bmatrix}$$

To find l_{ij} and u_{jk}

1st col.: $l_{i1} = a_{i1}$ **for** $i = 1$ **to** n.

2nd col.: $l_{i2} = a_{i2} - l_{i1} u_{12}$ **for** $i = 2$ **to** n.

LU Decomposition (2)

1st row: $$u_{1k} = \frac{a_{1k}}{l_{11}} \quad \text{for } k = 2 \text{ to } n$$

2nd row: $$u_{2k} = \frac{a_{2k} - l_{21} u_{1k}}{l_{22}}$$

(Eq. 1) $$l_{ij} = a_{ij} - \sum_{m=1}^{j-1} l_{im} u_{mj}, \quad n \geq i \geq j$$

(Eq. 2) $$u_{jk} = \frac{a_{jk} - \sum_{m=1}^{j-1} l_{jm} u_{mk}}{l_{jj}}, \quad k > j$$

Algorithm LU-Decomposition

1. **Set j = 1 goto 3.**

2. **Use (Eq. 1) to compute col. j in L, if j = n stop.**

3. **Use (Eq. 2) to compute row j in U.**

4. **Set j = j+1 goto 2.**

Solution Finding (1)

To solve $x = A^{-1}b$, we decompose $A = LU$ to obtain $LUx = b$.

Let $Ux = z$, we have then $Lz = b$

$$Lz = \begin{bmatrix} l_{11} & & & \\ l_{21} & l_{22} & & \\ \vdots & & \ddots & \\ l_{n1} & l_{n2} & \cdots & l_{nn} \end{bmatrix} \begin{bmatrix} z_1 \\ z_2 \\ \vdots \\ z_n \end{bmatrix} = \begin{bmatrix} b_1 \\ b_2 \\ \vdots \\ b_n \end{bmatrix}$$

$$\begin{aligned} l_{11}z_1 &= b_1 \\ l_{21}z_1 + l_{22}z_2 &= b_2 \\ l_{n1}z_1 + l_{n2}z_2 + \ldots + l_{nn}z_n &= b_n \end{aligned}$$

Solution Finding (2)

==> Forward Substitution:

$$z_1 = \frac{b_1}{l_{11}},$$

$$z_i = \frac{b_i - \sum_{m=1}^{i-1} l_{im} z_m}{l_{ii}}, \quad \text{for } i = 2 \ldots n$$

Now, we can solve x from Ux = z:

$$Ux = \begin{bmatrix} 1 & u_{12} & u_{13} & & u_{1n} \\ & 1 & u_{23} & & u_{2n} \\ & & 1 & \ldots & u_{3n} \\ & & & \ddots & \\ & & & & 1 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \\ x_3 \\ \vdots \\ x_n \end{bmatrix} = \begin{bmatrix} z_1 \\ z_2 \\ z_3 \\ \vdots \\ z_n \end{bmatrix} \qquad \begin{aligned} x_1 + u_{12}x_2 + \ldots + u_{1n}x_n &= z_1 \\ x_2 + \ldots + u_{2n}x_n &= z_2 \\ &\vdots \\ x_{n-1} + u_{n-1,n}x_n &= z_{n-1} \\ x_n &= z_n \end{aligned}$$

Solution Finding (3)

$$Ux = \begin{bmatrix} 1 & u_{12} & u_{13} & & u_{1n} \\ & 1 & u_{23} & & u_{2n} \\ & & 1 & \dots & u_{3n} \\ & & & \ddots & \\ & & & & 1 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \\ x_3 \\ \vdots \\ x_n \end{bmatrix} = \begin{bmatrix} z_1 \\ z_2 \\ z_3 \\ \vdots \\ z_n \end{bmatrix} \qquad \begin{array}{rl} x_1 + u_{12}x_2 + \dots + u_{1n}x_n & = z_1 \\ x_2 + \dots + u_{2n}x_n & = z_2 \\ \vdots & \\ x_{n-1} + u_{n-1,n}x_n & = z_{n-1} \\ x_n & = z_n \end{array}$$

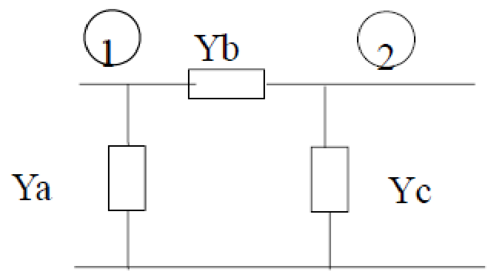
==> Backward Substitution:

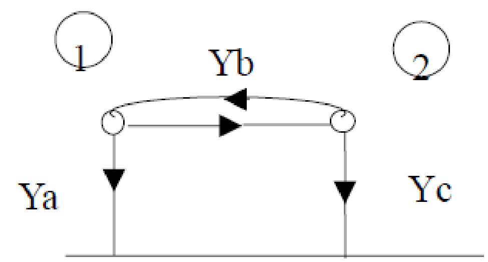
$$x_n = z_n$$

$$x_i = z_i - \sum_{m=i+1}^{n} u_{im} x_m, \quad \text{for } i = n-1, n-2, \dots, 1$$

Tree (Digraph) Representation of RLC-gm Network

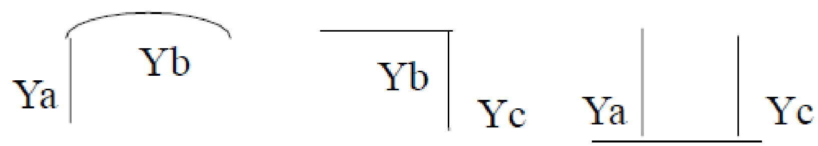
Use I = YV as example, where Y is a node-admittance matrix





*** Matrix representation:**

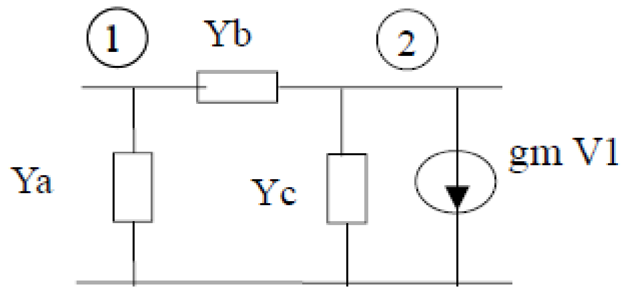
$$Y = \begin{bmatrix} Ya + Yb & -Yb \\ -Yb & Yb + Yc \end{bmatrix}$$

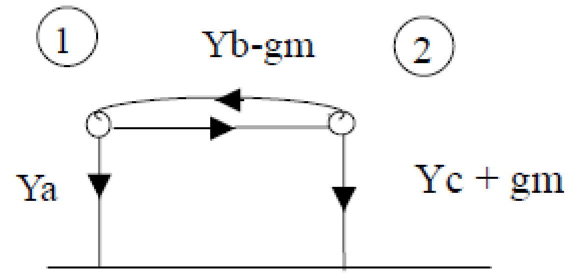


$$\begin{aligned} \Delta &= YaYb + YaYc + YbYb + YbYc - YbYb \\ &= YaYb + YaYc + YbYc \end{aligned}$$

*** Tree representation:** $\delta = YaYb + YbYc + YaYc$

Tree Representation Example (adding a current to Yc)



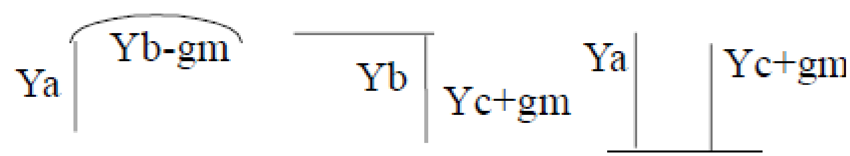


*** Matrix representation:**

$$Y = \begin{bmatrix} Ya + Yb & -Yb \\ -Yb + gm & Yb + Yc \end{bmatrix}$$

$$\begin{aligned} \Delta &= YaYb + YaYc + YbYb + YbYc - YbYb + Ybgm \\ &= YaYb + YaYc + YbYc + Ybgm \end{aligned}$$

*** Tree representation:**



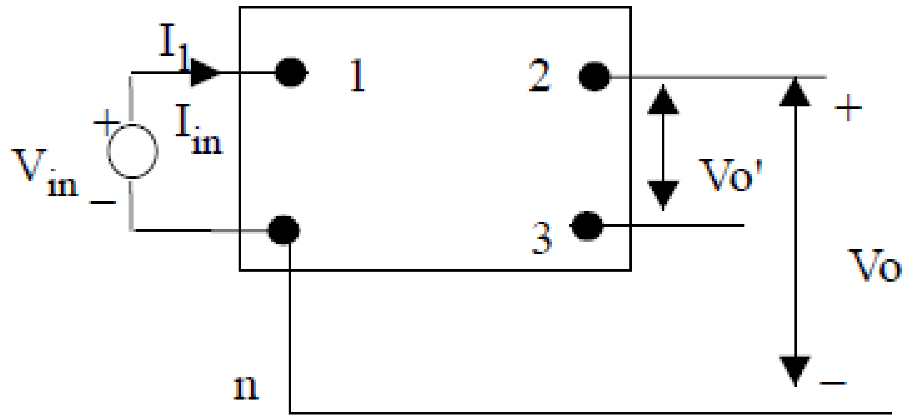
$\delta = YaYb + YbYc + YaYc + Yb\,gm$

*** Relationship between Δ and δ:** $\Delta_{jj} = \delta_{jj}$

$\Delta_{jk} = \delta_{jj} - \delta_{jk}$

Network with Voltage Source Input (1)

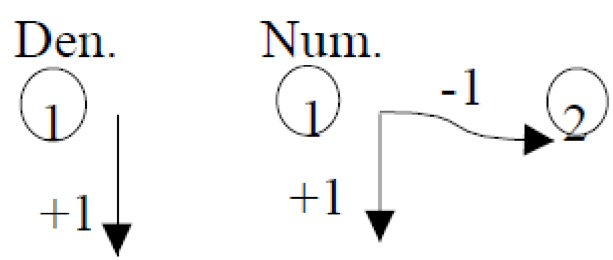
A. General Case (n: reference point)



(a1) Voltage Source cases:

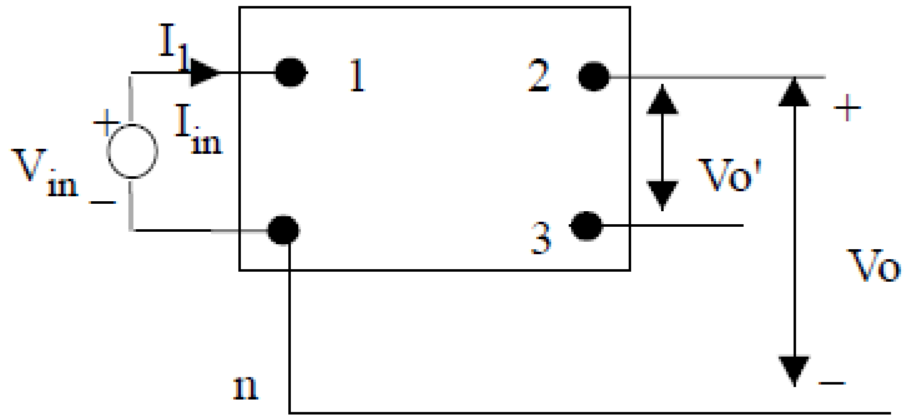
$$\frac{V_o}{V_{in}} = \frac{V_2}{V_1} = \frac{\Delta_{12}}{\Delta_{11}}$$

$$= \frac{\delta_{11} - \delta_{12}}{\delta_{11}} = \frac{\text{Numerator}}{\text{Denominator}}$$



Network with Voltage Source Input (2)

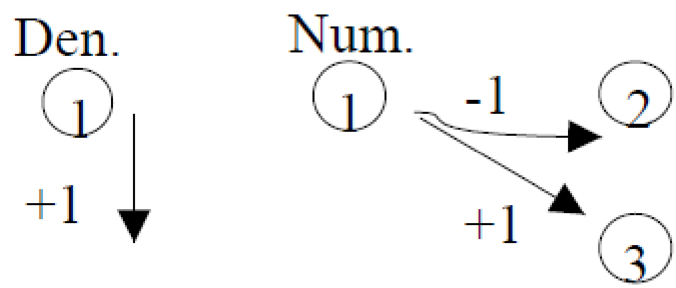
A. General Case (n: reference point)



(a2) Voltage Source cases (Cont.):

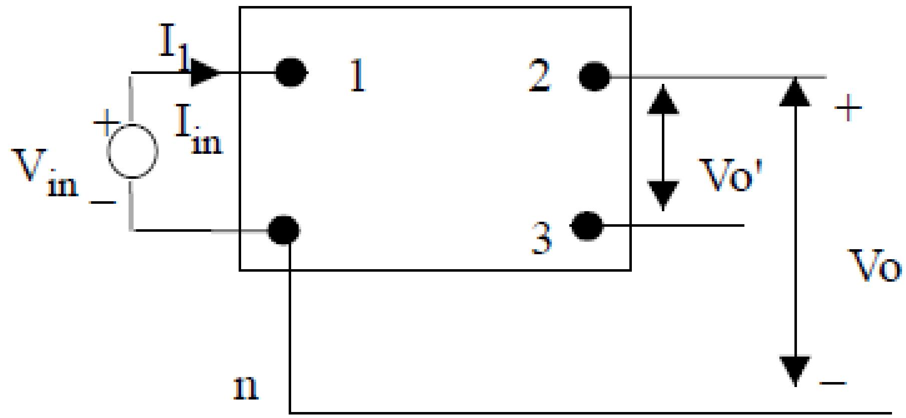
$$\frac{V_{o'}}{V_{in}} = \frac{V_2 - V_3}{V_1} = \frac{\Delta_{12} - \Delta_{13}}{\Delta_{11}}$$

$$= \frac{\delta_{13} - \delta_{12}}{\delta_{11}} = \frac{\text{Num.}}{\text{Den.}}$$



Network with Current Source Input (1)

A. General Case (n: reference point)



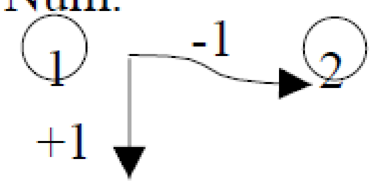
(b1) Current Source Cases:

$$\frac{V_o}{I_{in}} = \frac{V_2}{I_1} = \frac{\Delta_{12}}{\Delta}$$

$$= \frac{\delta_{11} - \delta_{12}}{\delta} = \frac{\text{Num.}}{\text{Den.}}$$

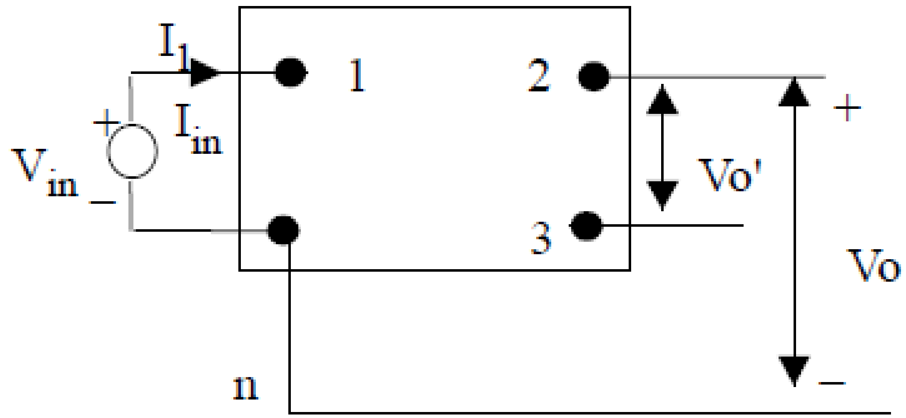
Den. Num.

Whole Tree



Network with Current Source Input (2)

A. General Case (n: reference point)



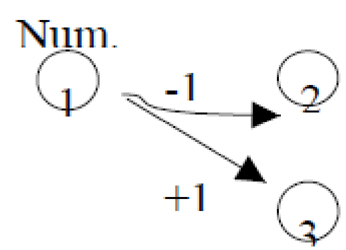
(b2) Current Source cases (Cont.):

$$\frac{V_{o'}}{I_{in}} = \frac{V_2 - V_3}{I_1} = \frac{\Delta_{12} - \Delta_{13}}{\Delta}$$

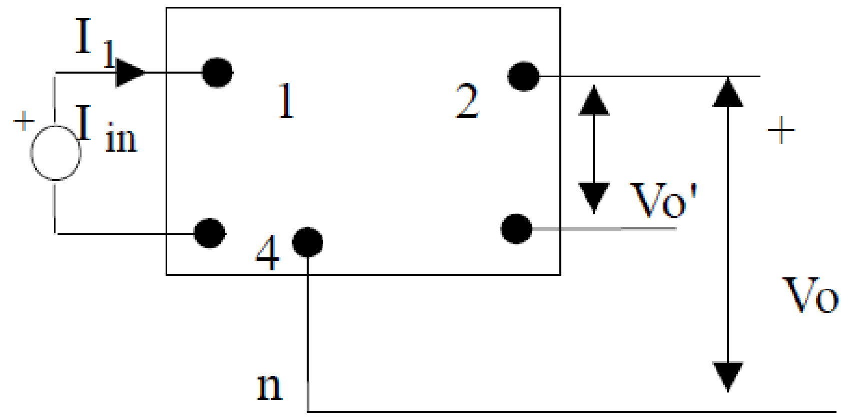
$$= \frac{\delta_{13} - \delta_{12}}{\delta} = \frac{\text{Num.}}{\text{Den.}}$$

Den.

Whole Tree

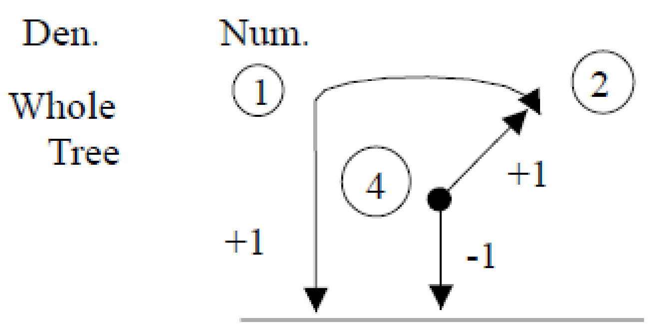


B. Special Case with Current Source not Grounded

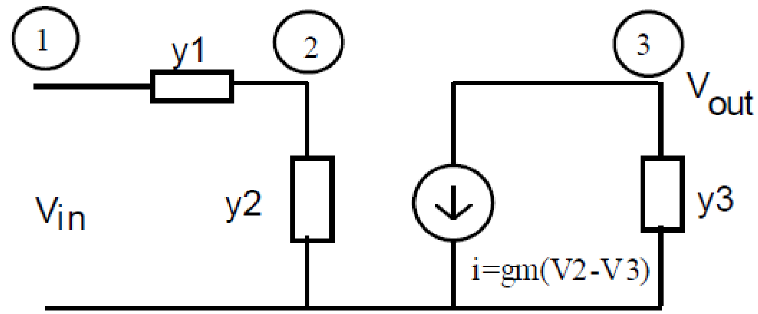


$$\frac{V_o}{I_{in}} = \frac{V_2}{I_{14}} = \frac{\Delta_{12} - \Delta_{42}}{\Delta}$$

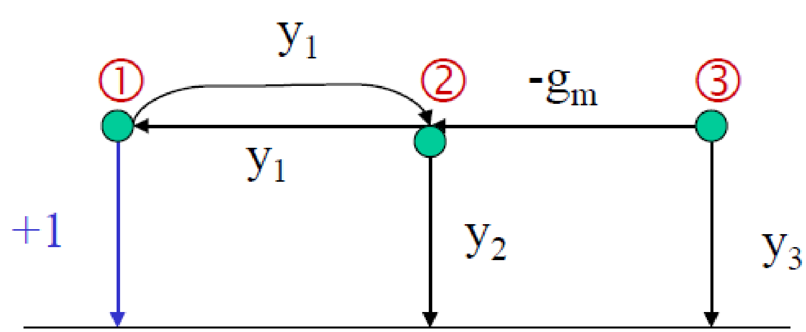
$$= \frac{\delta_{11} - \delta_{12} - (\delta_{44} - \delta_{42})}{\delta} = \frac{\text{Num.}}{\text{Den.}}$$



Example: Use digraph to find Vout / Vin (1)



Denominator: voltage source ⇒ add a +1 branch to node ①-0, and we have:

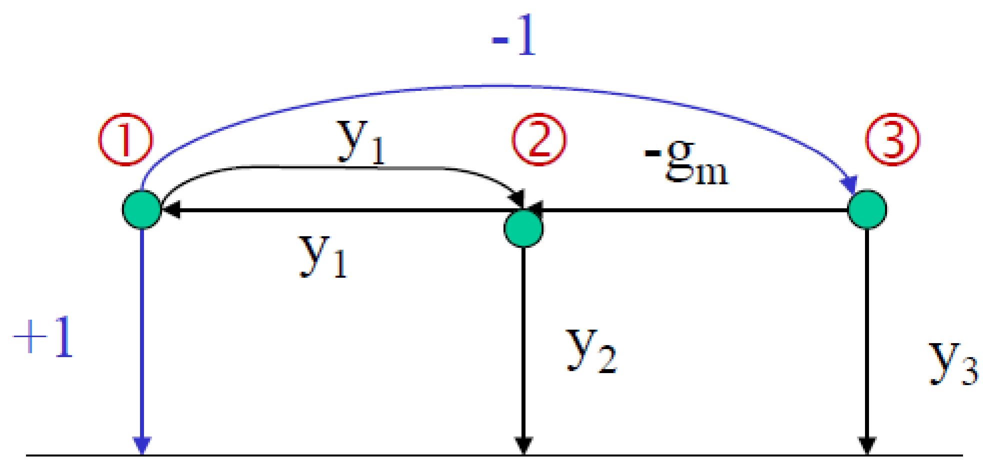


$$V_{in} = \Delta_{11} = \delta_{11} = - y_1 g_m + y_1 y_3 + y_2 y_3 - y_2 g_m \qquad \text{(a)}$$

Example: Use digraph to find Vout / Vin (2)

Numerator: add +1, -1 branches to nodes ①-0, ①-③,

and we have:



$$\delta_{11} = -y_1 g_m + y_1 y_3 + y_2 y_3 - y_2 g_m \qquad \text{(b)}$$

$$\delta_{13} = y_2 g_m - y_2 y_3 - \quad y_1 y_3 \qquad \text{(c)}$$

$$V_{out} = \Delta_{13} = \delta_{11} - \delta_{13} = \text{(b)} - \text{(c)} = -y_1 g_m \qquad \text{(d)}$$

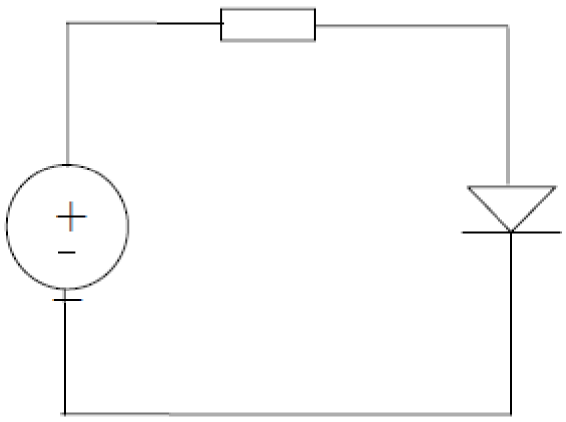
$$\Rightarrow \quad V_{out} / V_{in} = \text{(d)} / \text{(a)} = -y_1 g_m / (-y_1 g_m + y_1 y_3 + y_2 y_3 - y_2 g_m)$$

Newton-Raphson Algorithm and its Application (1)

Solve for $f(x) = 0$, we have

$$\Delta x = x^{j+1} - x^{j} = \frac{-f(x^{j})}{f'(x^{j})}$$

*** For a diode, we have:**



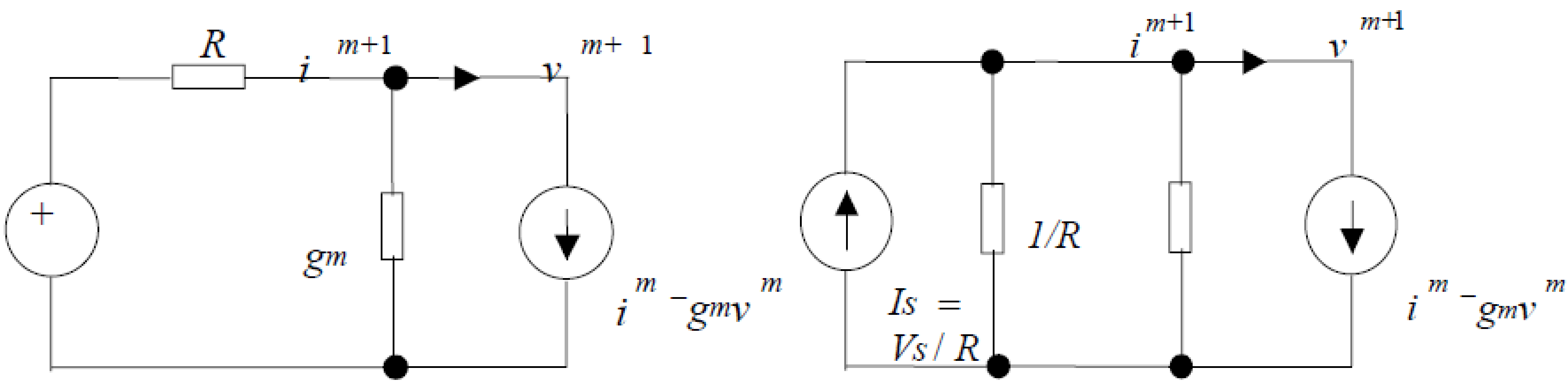
$$i^{m} = I_{D} = I_{0}(e^{v^{m}/v_{T}} - 1)$$

$$g_{m} = \frac{d\, i^{m}}{d\, v} = \frac{I_{s}}{v_{T}}(e^{v^{m}/v_{T}})$$

Newton-Raphson Algorithm and its Application (2)

*** Use N-R method to find i^{m+1}, v^{m+1} and its companion (or associated discrete circuit) models:**

(Eq. 3) $$i^{m+1} = (i^m - g_m v^m) + g_m v^{m+1}$$



(Eq. 4) $$v^{m+1} = \frac{I_s - (i^m - g_m v^m)}{1/R + g_m}$$

Notes

1. If given only i^m, use $i^m = I_0(e^{v^m/v_T} - 1)$ to find v^m.

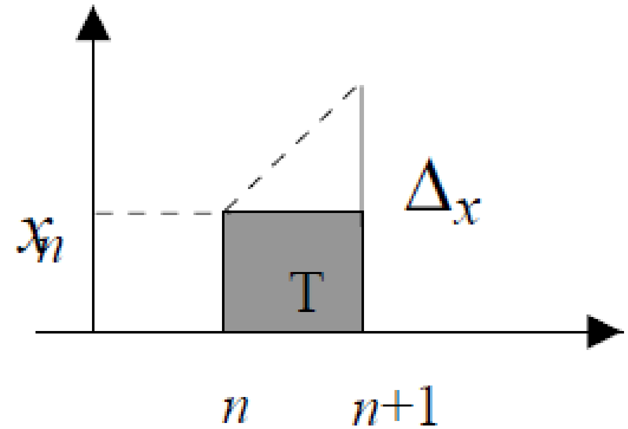
2. If we want i^{m+1}, find v^{m+1} first by (Eq. 4), then use (Eq. 3) to find i^{m+1}.

Numerical Integration (1)

(1) Forward Euler (FE): slope

$$\dot{x}_n = \frac{\Delta x}{T}$$

$$x_{n+1} = x_n + T\dot{x}_n$$



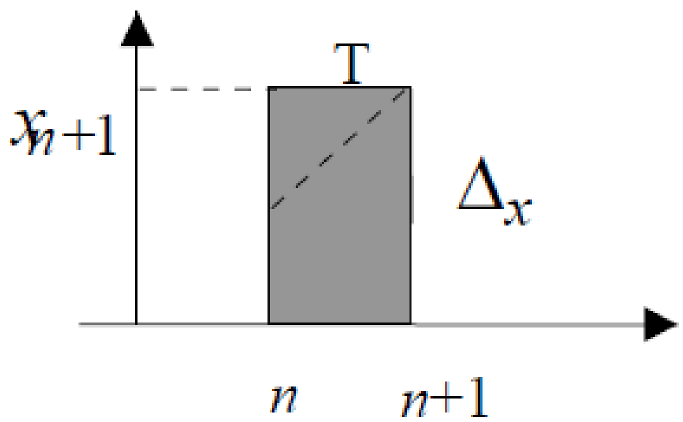
*** ST-plane of FE:** $ST = e^{j\theta} - 1$

Numerical Integration (2)

(2) Backward Euler (BE): slope

$$\dot{x}_{n+1} = \frac{\Delta x}{T}$$

$$x_{n+1} = x_n + T\dot{x}_{n+1}$$



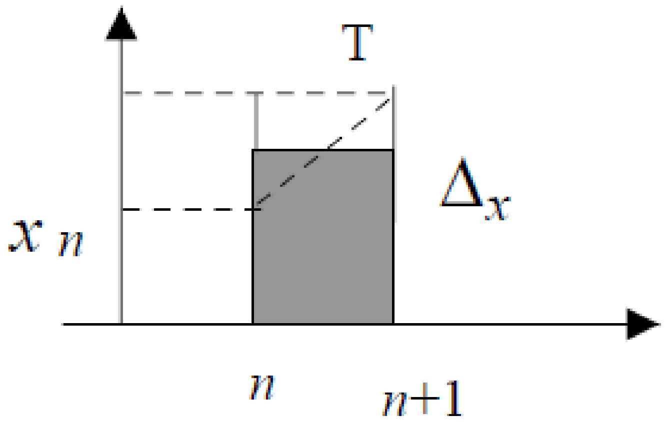
*** ST plane of BE:** $ST = 1 - e^{j\theta}$

Numerical Integration (3)

(3) Trapezoid (Trap): slope

$$\frac{\dot{x}_n + \dot{x}_{n+1}}{2} = \frac{\Delta x}{T}$$

$$x_{n+1} = x_n + \frac{T}{2}(\dot{x}_n + \dot{x}_{n+1})$$



*** ST plane of Trap:**

$$ST = \frac{2(e^{j\theta} - 1)}{e^{j\theta} + 1} = 2j\tan\frac{\theta}{2}$$

Now, we have:

$$x_n = e^{At_n} = x(t_n)$$

$$x_{n+1} = x(t_n + T) = e^{A(t_n+T)} = e^{AT} x_n = z x_n$$

$$x_{n-1} = x(t_n - T) = e^{A(t_n-T)} = e^{-AT} x_n = z^{-1} x_n$$

$$\dot{x}_n = A x_n = A e^{At_n}$$

$$\dot{x}_{n+1} = A x_{n+1}$$

(Eq. 5)

Procedure to Find ROC (Stability Problem)

1. **Take z-tx of the system differential equation by (Eq. 5)**

   for $x_n = e^{At_n},\ x_{n+1} = e^{At} x_n,\ \dot{x}_n = Ax_n,\ \dot{x}_{n+1} = Ax_{n+1}, \ldots$

2. **AT ==> ST, get Eq. for** $ST = \dfrac{C(z+b)}{(z+a)} = \dfrac{C(e^{j\theta}+b)}{(e^{j\theta}+a)}$

3. **Plot on ST plane for** $e^{j\theta} = 1, -1, j, -j, 0$ **to find the curve and Region of Convergence (ROC).**

   Or, if we can rearrange the ST formula to get $|z| = (a+ST)/(b+ST)$**. By substituting ST with** $x + jy$**, and by setting** $|z| = (a+x+jy)/(b+x+jy) = 1$**, we can find the equation for ST plane and its ROC too.**

To find Coefficients for Best Accuracy (1)

Use (Eq. 5) and $e^{AT} = (1 + AT + \frac{AT^2}{2} + \frac{AT^3}{3!} + \ldots),$ we can get the answer by comparison of coefficients.

* **For Forward Euler:**

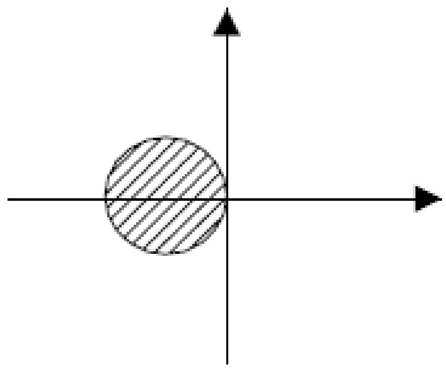
1. Since we have $x_{n+1} = x_n + T\dot{x}_n,$

   that is, $zx_n = x_n + T(Ax_n)$, $AT = z - 1$

2. Its ST-plane: $ST = e^{j\theta} - 1 = z - 1$; $z = 1 + S_i T$

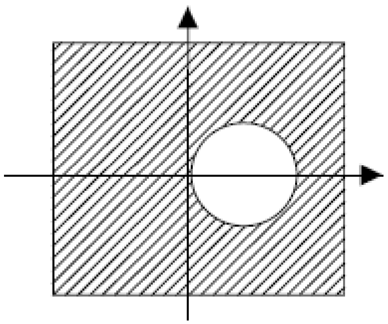
3. Thus, $|z| = |1 + S_i T| = |1 + x + jy| = 1$, or $(x+1)^2 + y^2 = 1$,

   we can then plot the ROC as:

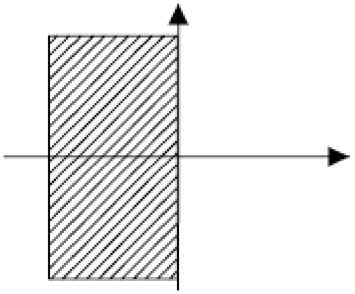


To find Coefficients for Best Accuracy (2)

* **For Backward Euler:** $ST = 1 - e^{-j\theta}$,



* **For Trap:** $$ST = \frac{2(e^{j\theta} - 1)}{e^{j\theta} + 1} = 2j\tan\frac{\theta}{2}$$



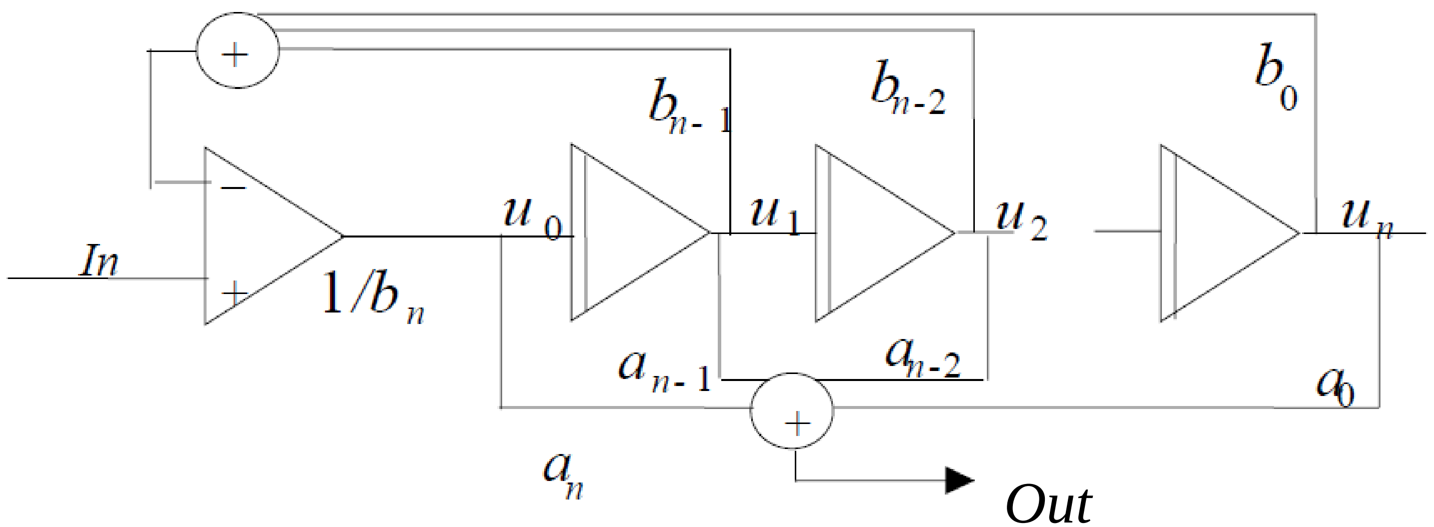
Time Response (1)

For a general system, by defining $u_k = u_0/s^k$, we have:

$$\frac{Out}{In} = \frac{a_0 + a_1 s + a_2 s^2 + \ldots + a_n s^n}{b_0 + b_1 s + b_2 s^2 + \ldots + b_n s^n}$$

$$= \frac{a_0 u_n + a_1 u_{n-1} + a_2 u_{n-2} + \ldots + a_n u_0}{b_0 u_n + b_1 u_{n-1} + b_2 u_{n-2} + \ldots + b_n u_0}$$

We have also:



Time Response (2)

$$In = b_n u_0 + \sum_{k=1}^{n} b_{n-k} u_k, \quad u_0 = \left(In - \sum_{k=1}^{n} b_{n-k} u_k \right) \Big/ b_n$$

If we use BE:

$$u_{k,n+1} = u_{k,n} + T \cdot u_{k-1,n+1}$$

$$= u_{k,n} + \sum_{j=1}^{k-1} (T)^j \cdot u_{k-j,n} + (T)^k \cdot u_{0,n+1}$$

we can then solve for BE the following (Eq. 6):

$$u_{0,n+1} = \frac{In - \sum_{k=1}^{n} \left\{ b_{n-k} \left[u_{k,n} + \sum_{j=1}^{k-1} (T)^j \cdot u_{k-j,n} \right] \right\}}{\sum_{k=0}^{n} (T)^k \cdot b_{n-k}}$$

Time Response (3)

* **Use Trapezoid, we have:**

$$u_{k,n+1} = u_{k,n} + \frac{T}{2}\cdot[u_{k-1,n} + u_{k-1,n+1}]$$

$$= u_{k,n} + 2\sum_{j=1}^{k-1}\left(\frac{T}{2}\right)^{j}\cdot u_{k-j,n} + \left(\frac{T}{2}\right)^{k}\cdot[u_{0,n} + u_{0,n+1}]$$

we can solve for Trap the following (Eq. 7):

$$u_{0,n+1} = \frac{In - \sum_{k=1}^{n}\left\{b_{n-k}\left[u_{k,n} + 2\sum_{j=1}^{k-1}\left(\frac{T}{2}\right)^{j}\cdot u_{k-j,n} + \left(\frac{T}{2}\right)^{k}\cdot u_{0,n}\right]\right\}}{\sum_{k=0}^{n}\left(\frac{T}{2}\right)^{k}\cdot b_{n-k}}$$

Linear Multi-Step (LMS) Algorithm

Step 0. Set up initial values: $u_{0,n} = In(0)/b_n$,

$u_{k,n} = 0$, **for** $k = 1, \ldots, n$

Step 1. Find new $u_{0,\,n+1}$ **using (Eq. 6) or (Eq. 7).**

Step 2. Find all new u_k **by using old** u_k **and** u_{k-1}

$$u_{k,n+1} = u_{k,n} + T \cdot u_{k-1,n+1} \quad \text{or}$$

$$u_{k,n+1} = u_{k,n} + \frac{T}{2} \cdot [u_{k-1,n} + u_{k-1,n+1}]$$

Step 3. Find output time function value by

$$Out = \sum_{k=0}^{n} a_{n-k} u_{k,n+1}$$

Step 4. Set $u_{k,n} = u_{k,n+1}$, **for** $k = 0, \ldots, n$ **then goto Step 1.**

Frequency Response

$$F(s) = \frac{a_0 + a_1 s + a_2 s^2 + \ldots + a_n s^n}{b_0 + b_1 s + b_2 s^2 + \ldots + b_n s^n}$$

Substitute $s = jw$ into it, plot Frequency in Hz, Amplitude in DB, and Phase in Deg.

$$DB = 10 \log \left| \frac{V_{in}}{V_{out}} \right|^2 = 20 \log \left| \frac{V_{in}}{V_{out}} \right|$$

$$A + jB = \sqrt{A^2 + B^2} \angle \theta$$